

# Annex B: Sample Problem

(informative)

## B.1 Purpose

The purpose of this annex is to illustrate how SysML can support the specification, analysis, and design of a system using some of the basic features of the language.

## B.2 Scope

The scope of this example is to provide at least one diagram for each SysML diagram type. The intent is to select simplified fragments of the problem to illustrate how the diagrams can be applied, and also demonstrate some of the possible inter-relationships among the model elements in the different diagrams. The sample problem does not highlight all of the features of the language. The reader should refer to the individual chapters for more detailed features of the language. The diagrams selected for representing a particular aspect of the model, and the ordering of the diagrams are intended to be representative of applying a typical systems engineering process, but this will vary depending on the specific process and methodology that is used.

## B.3 Problem Summary

The sample problem describes the use of SysML as it applies to the development of an automobile, in particular a Hybrid gas/electric powered Sport Utility Vehicle (SUV). This problem is interesting in that it has inherently conflicting requirements, viz. desire for fuel efficiency, but also desire for large cargo carrying capacity and off-road capability. Technical accuracy and the feasibility of the actual solution proposed were not high priorities. This sample problem focuses on design decisions surrounding the power subsystem of the hybrid SUV; the requirements, performance analyses, structure, and behavior.

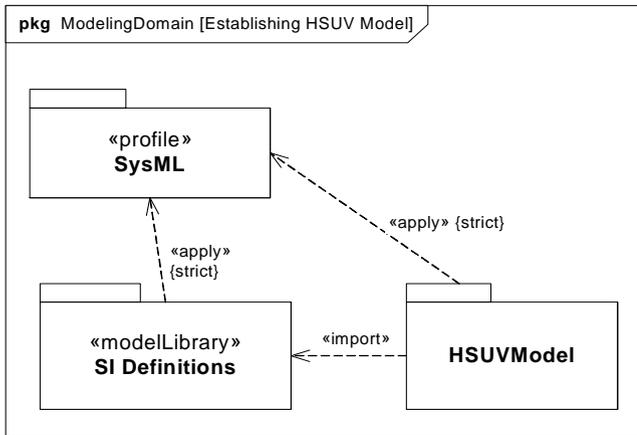
This annex is structured to show each diagram in the context of how it might be used on such an example problem. The first section shows SysML diagrams as they might be used to establish the system context; establishing system boundaries, and top level use cases. The next section is provided to show how SysML diagrams can be used to analyze top level system behavior, using sequence diagrams and state machine diagrams. The following section focuses on use of SysML diagrams for capturing and deriving requirements, using diagrams and tables. A section is provided to illustrate how SysML is used to depict system structure, including block hierarchy and part relationships. The relationship of various system parameters, performance constraints, analyses, and timing diagrams are illustrated in the next section. A section is then dedicated to illustrating definition and depiction of interfaces and flows in a structural context. The final section focuses on detailed behavior modeling, functional and flow allocation.

## B.4 Diagrams

### B.4.1 Package Overview (Structure of the Sample Model)

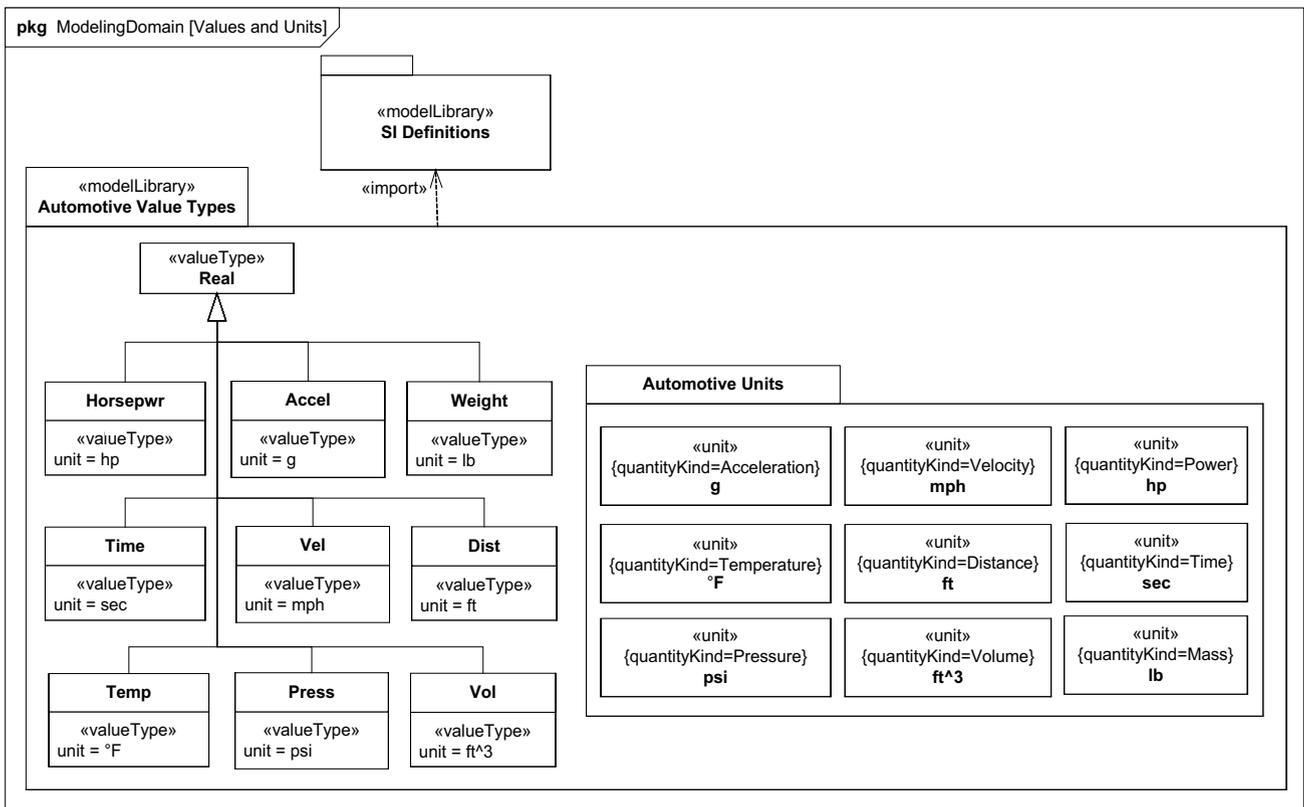
#### B.4.1.1 Package Diagram - Applying the SysML Profile

As shown in Figure B.1, the HSUVModel is a package that represents the user model. The SysML Profile must be applied to this package in order to include stereotypes from the profile. The HSUVModel may also require model libraries, such as the SI Units Types model library. The model libraries must be imported into the user model as indicated.



**Figure B.1 - Establishing the User Model by Importing and Applying SysML Profile & Model Library (Package Diagram)**

Figure B.2 details the specification of units and valueTypes employed in this sample problem.



**Figure B.2 - Defining valueTypes and units to be Used in the Sample Problem**

#### B.4.1.2 Package Diagram - Showing Package Structure of the Model

The package diagram (Figure B.3) shows the structure of the model used to evaluate the sample problem. Model elements are contained in packages, and relationships between packages (or specific model elements) are shown on this diagram. The

relationship between the views (OperationalView and PerformanceView) and the rest of the user model are explicitly expressed using the «import» relationship. Note that the «view» models contain no model elements of their own, and that changes to the model in other packages are automatically updated in the Operational and Performance Views.

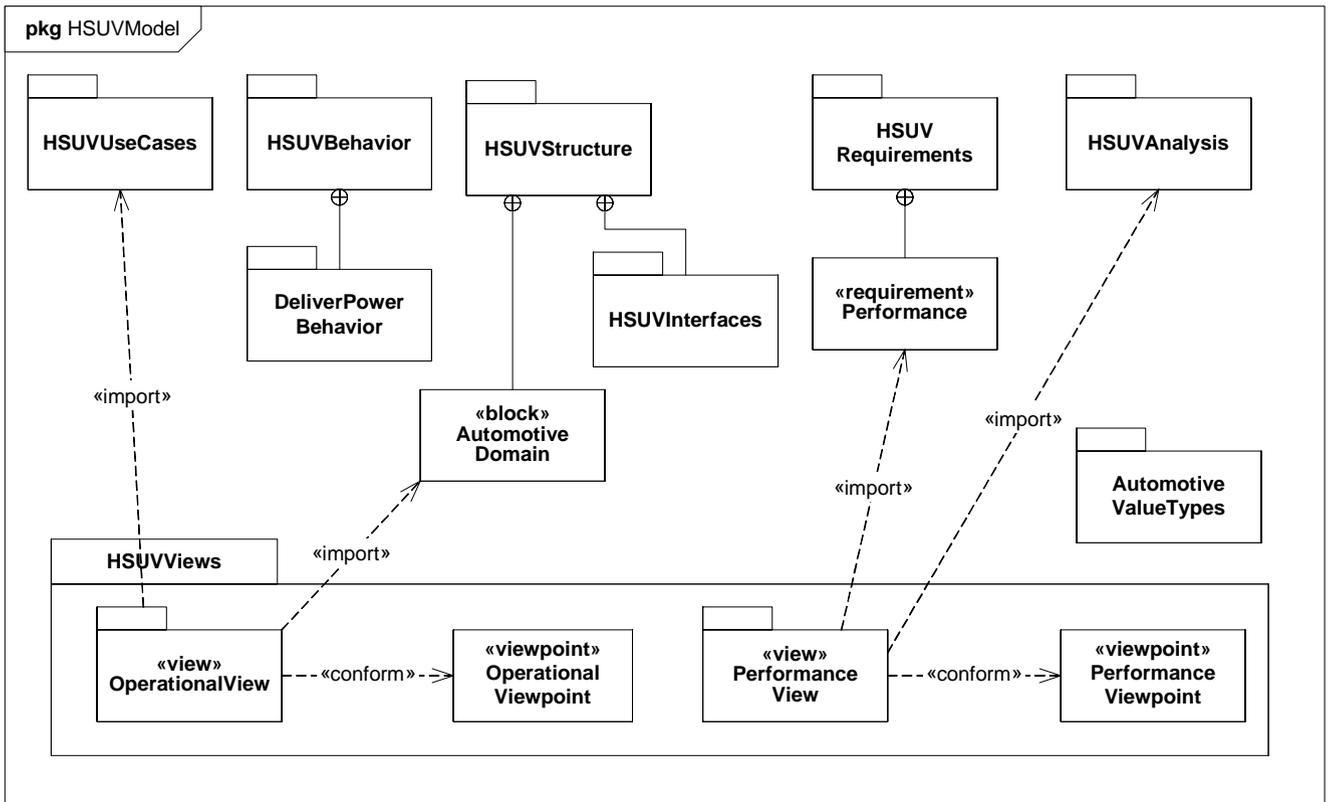
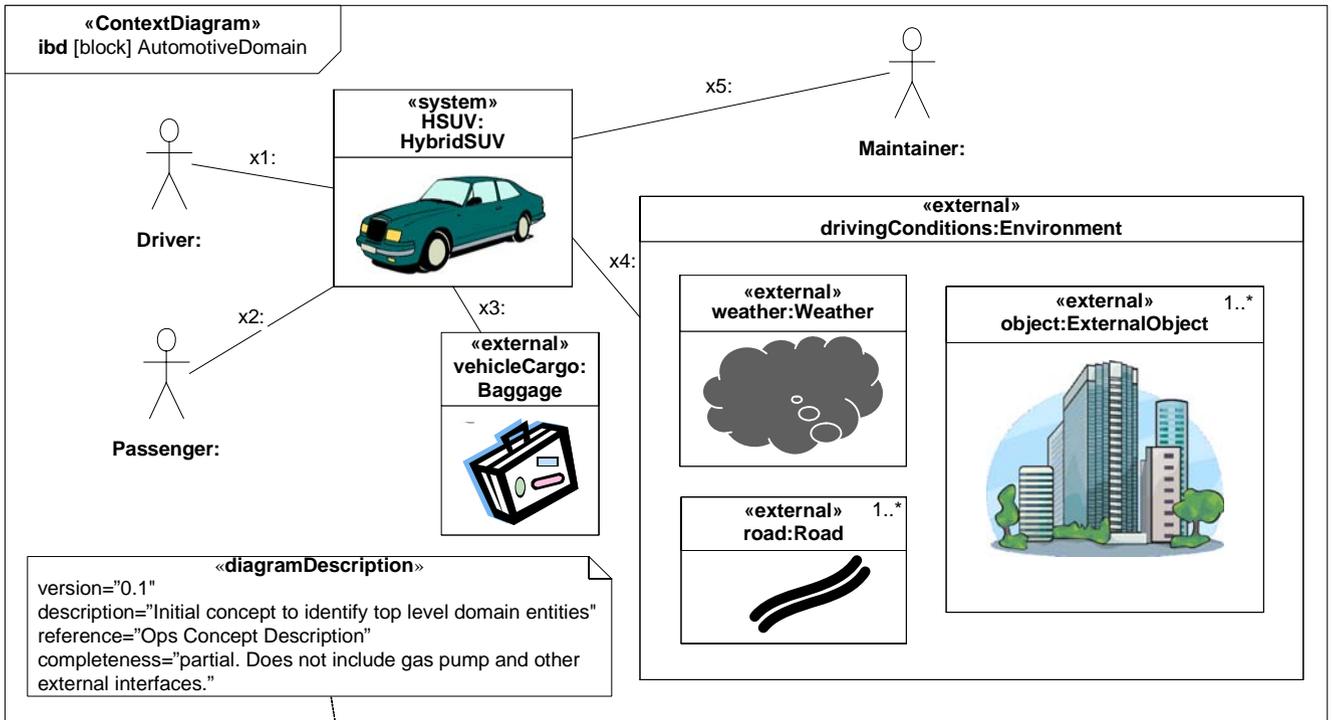


Figure B.3 - Establishing Structure of the User Model using Packages and Views (Package Diagram)

## B.4.2 Setting the Context (Boundaries and Use Cases)

### B.4.2.1 Internal Block Diagram - Setting Context

The term “context diagram,” in Figure B.4, refers to a user-defined usage of an internal block diagram, which depicts some of the top-level entities in the overall enterprise and their relationships. The diagram usage enables the modeler or methodologist to specify a unique usage of a SysML diagram type using the extension mechanism described in Annex A: Diagrams. The entities are conceptual in nature during the initial phase of development, but will be refined as part of the development process. The «system» and «external» stereotypes are user defined, not specified in SysML, but help the modeler to identify the system of interest relative to its environment. Each model element depicted may include a graphical icon to help convey its intended meaning. The spatial relationship of the entities on the diagram sometimes conveys understanding as well, although this is not specifically captured in the semantics. Also, a background such as a map can be included to provide additional context. The associations among the classes may represent abstract conceptual relationships among the entities, which would be refined in subsequent diagrams. Note how the relationships in this diagram are also reflected in the Automotive Domain Model Block Definition Diagram, Figure B.15.



**Figure B.4 - Establishing the Context of the Hybrid SUV System using a User-Defined Context Diagram. (Internal Block Diagram) Completeness of Diagram Noted in Diagram Description**

### B.4.2.2 Use Case Diagram - Top Level Use Cases

The use case diagram for “Drive Vehicle” in Figure B.5 depicts the drive vehicle usage of the vehicle system. The subject (HybridSUV) and the actors (Driver, Registered Owner, Maintainer, Insurance Company, DMV) interact to realize the use case.

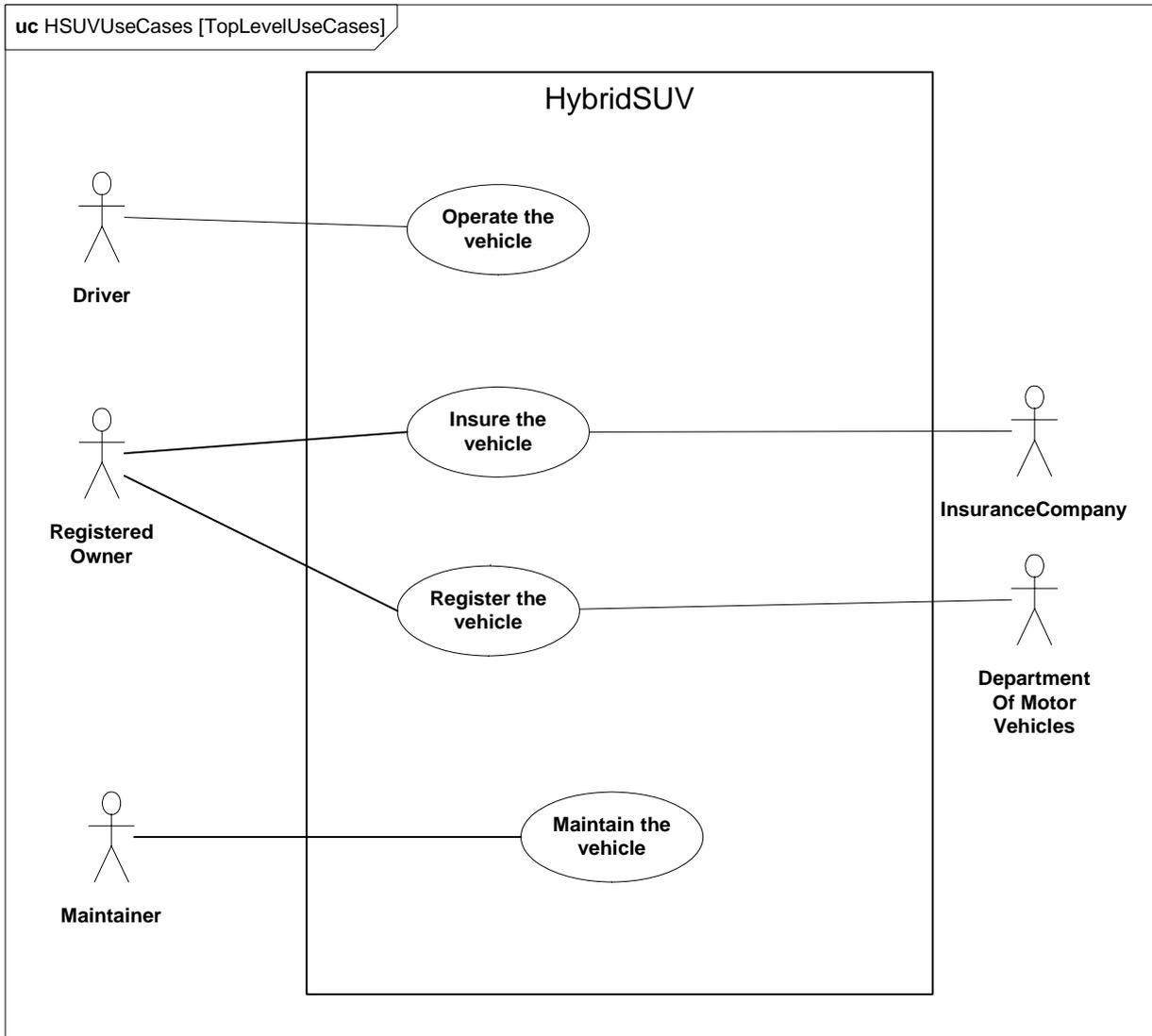


Figure B.5 - Establishing Top Level Use Cases for the Hybrid SUV (Use Case Diagram)

### B.4.2.3 Use Case Diagram - Operational Use Cases

Goal-level Use Cases associated with “Operate the Vehicle” are depicted in the following diagram. These use cases help flesh out the specific kind of goals associated with driving and parking the vehicle. Maintenance, registration, and insurance of the vehicle would be covered under a separate set of goal-oriented use cases.

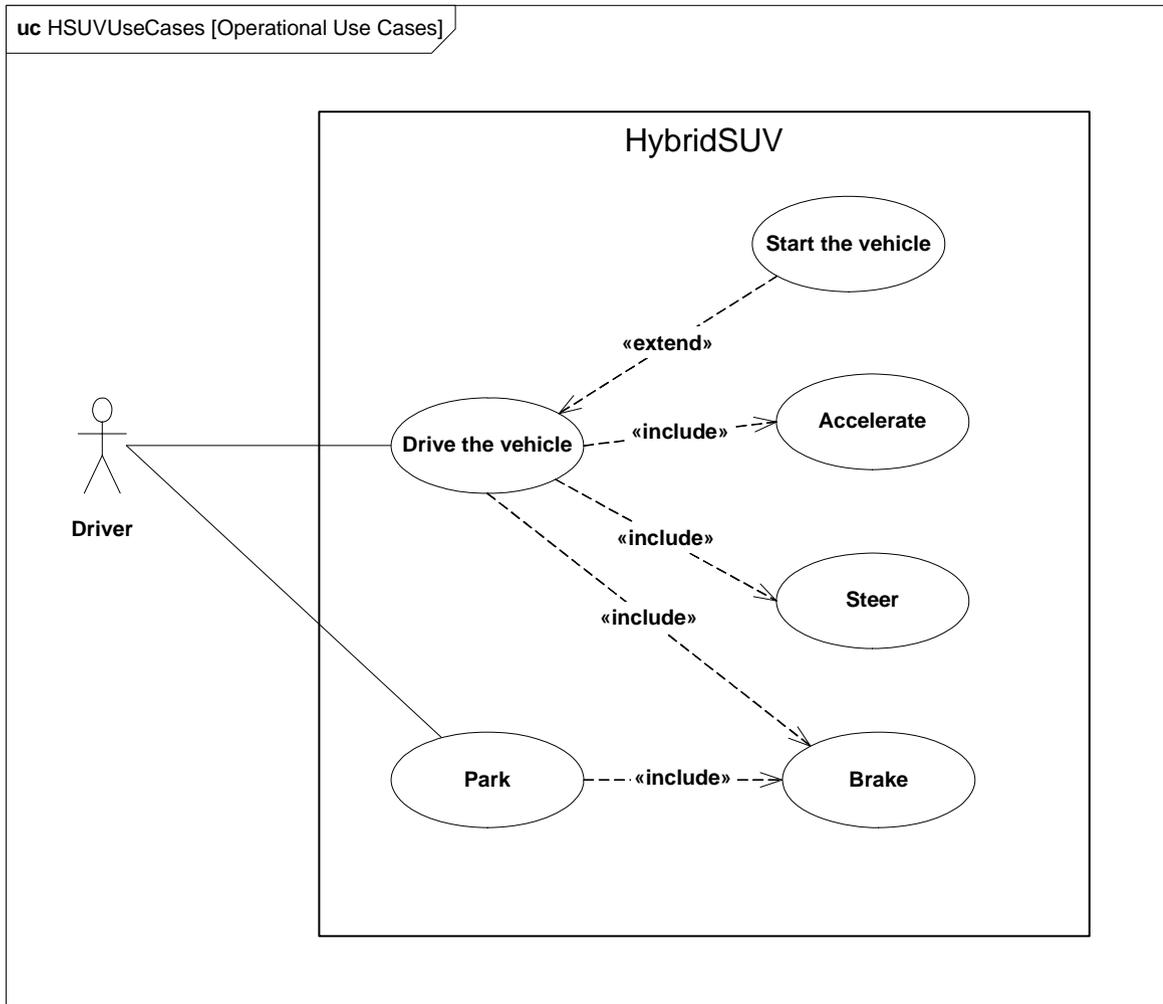


Figure B.6 - Establishing Operational Use Cases for “Drive the Vehicle” (Use Case Diagram)

## B.4.3 Elaborating Behavior (Sequence and State Machine Diagrams)

### B.4.3.1 Sequence Diagram - Drive Black Box

Figure B.7 shows the interactions between driver and vehicle that are necessary for the “Drive the Vehicle” Use Case. This diagram represents the “DriveBlackBox” interaction, with is owned by the AutomotiveDomain block. “BlackBox” for the purpose of this example, refers to how the subject system (HybridSUV block) interacts only with outside elements, without revealing any interior detail.

The conditions for each alternative in the alt controlSpeed section are expressed in OCL, and relate to the states of the HybridSUV block, as shown in Figure B.8.

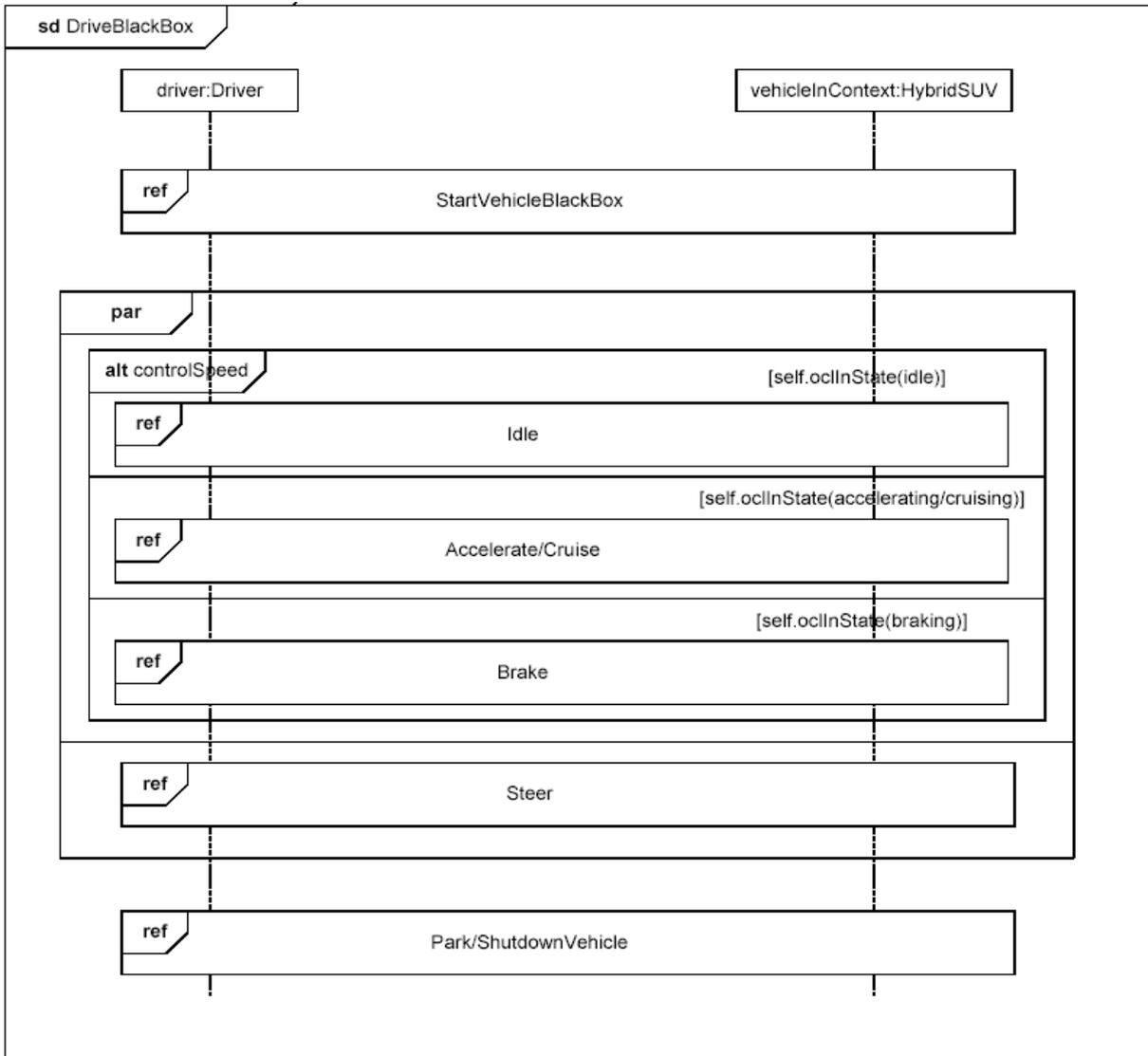


Figure B.7 - Elaborating Black Box Behavior for the “Drive the Vehicle” Use Case (Sequence Diagram)

### B.4.3.2 State Machine Diagram - HSUV Operational States

Figure B.8 depicts the operational states of the HSUV block, via a State Machine named “HSUVOperationalStates.” Note that this state machine was developed in conjunction with the DriveBlackBox interaction in Figure B.7. Also note that this state machine refines the requirement “PowerSourceManagement,” which will be elaborated in the requirements section of this sample problem. This diagram expresses only the nominal states. Exception states, like “acceleratorFailure,” are not expressed on this diagram.

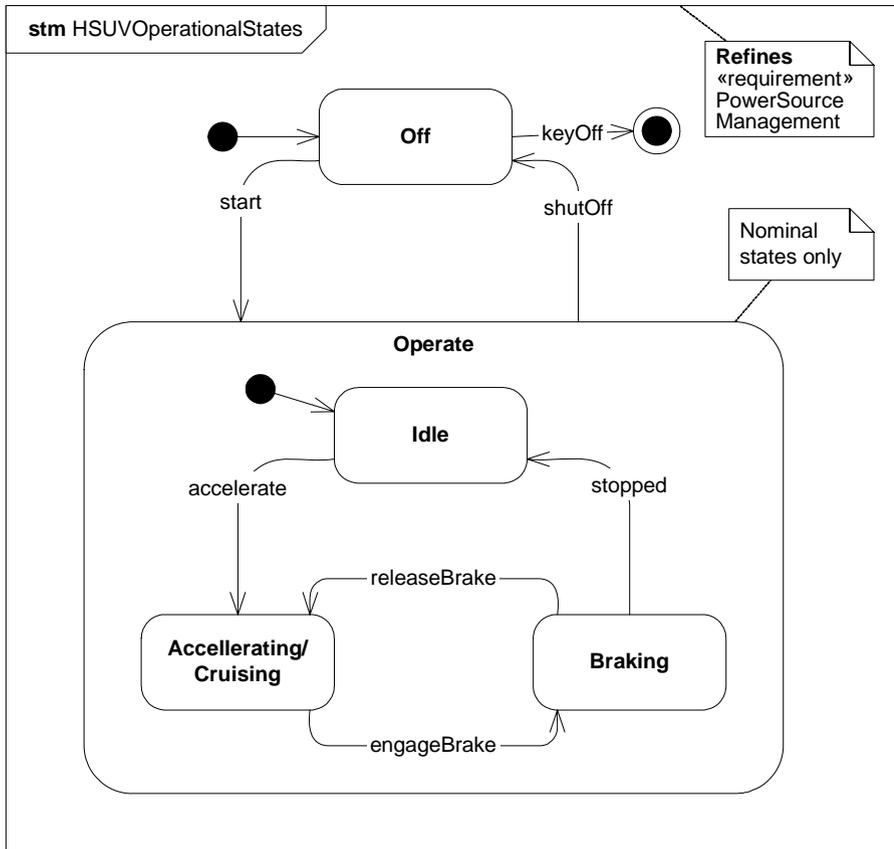


Figure B.8 - Finite State Machine Associated with "Drive the Vehicle" (State Machine Diagram)

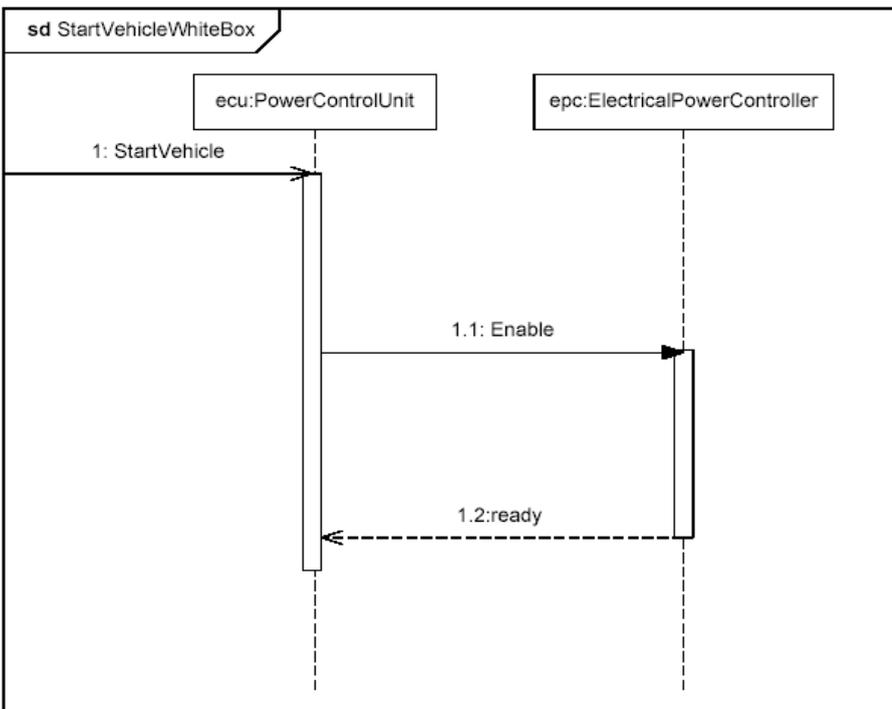
### B.4.3.3 Sequence Diagram - Start Vehicle Black Box & White Box

Figure B.9 shows a "black box" interaction, but references "StartVehicleWhiteBox" (Figure B.10), which will decompose the lifelines within the context of the HybridSUV block.



**Figure B.9 - Black Box Interaction for “StartVehicle,” referencing White Box Interaction (Sequence Diagram)**

The lifelines on Figure B.10 (“whitebox” sequence diagram) need to come from the Power System decomposition. This now begins to consider parts contained in the HybridSUV block.



**Figure B.10 - White Box Interaction for “StartVehicle” (Sequence Diagram)**

## B.4.4 Establishing Requirements (Requirements Diagrams and Tables)

### B.4.4.1 Requirement Diagram - HSUV Requirement Hierarchy

The vehicle system specification contains many text based requirements. A few requirements are highlighted in Figure B.11, including the requirement for the vehicle to pass emissions standards, which is expanded for illustration purposes. The containment (cross hair) relationship, for purposes of this example, refers to the practice of decomposing a complex requirement into simpler, single requirements.

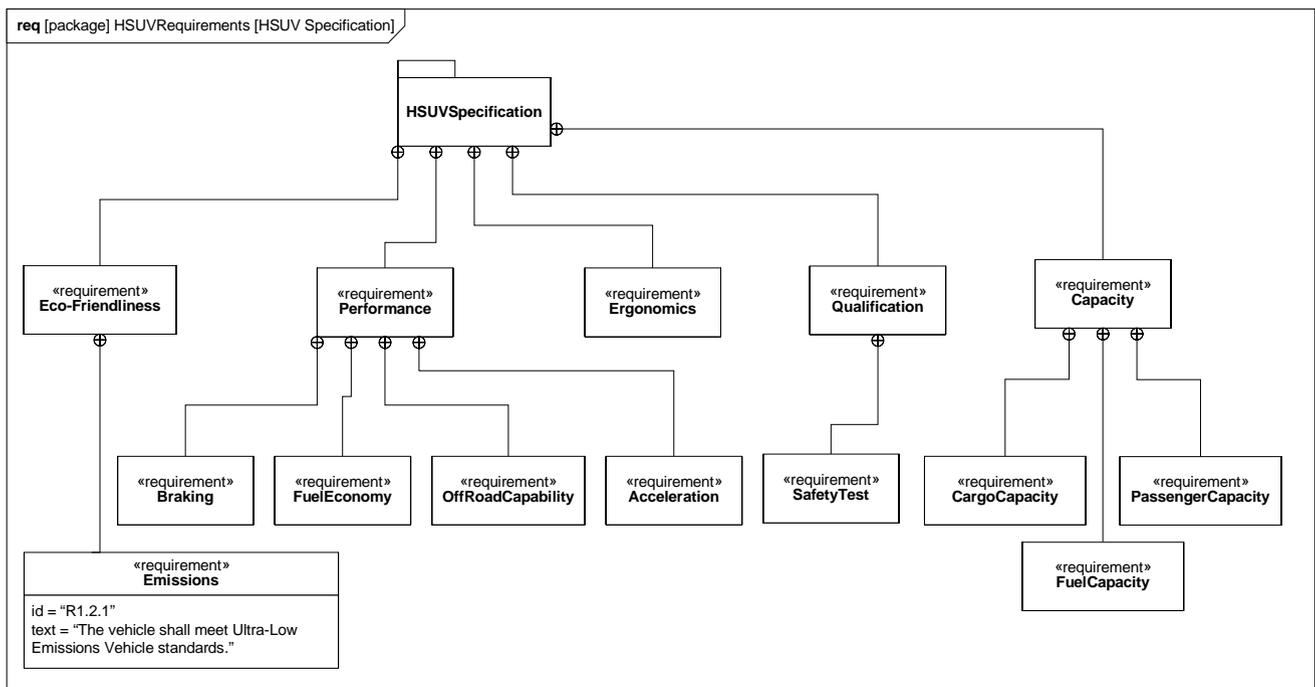


Figure B.11 - Establishing HSUV Requirements Hierarchy (containment) - (Requirements Diagram)

### B.4.4.2 Requirement Diagram - Derived Requirements

Figure B.12 shows a set of requirements derived from the lowest tier requirements in the HSUV specification. Derived requirements, for the purpose of this example, express the concepts of requirements in the HSUVSpecification in a manner that specifically relates them to the HSUV system. Various other model elements may be necessary to help develop a derived requirement, and these model element may be related by a «refinedBy» relationship. Note how PowerSourceManagement is “RefinedBy” the HSUVOperationalStates model (Figure B.8). Note also that rationale can be attached to the «deriveReq» relationship. In this case, rationale is provided by a referenced document “Hybrid Design Guidance.”

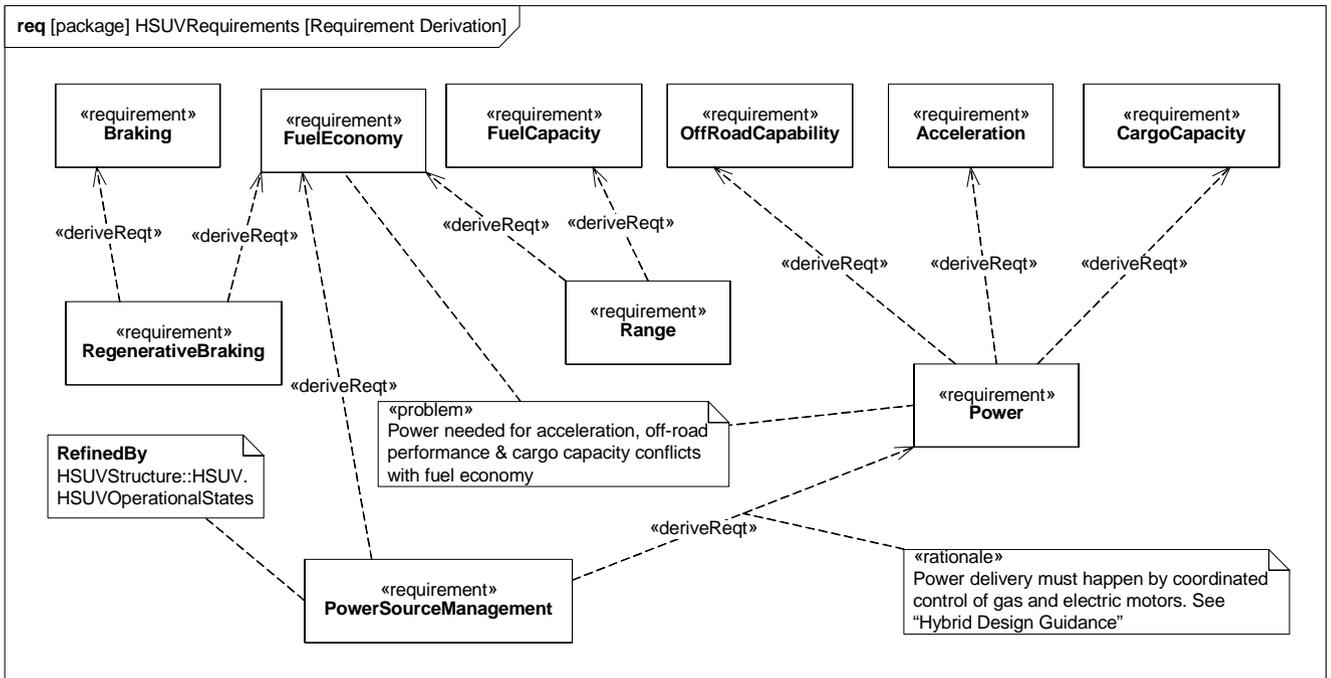


Figure B.12 - Establishing Derived Requirements and Rationale from Lowest Tier of Requirements Hierarchy (Requirements Diagram)

#### B.4.4.3 Requirement Diagram - Acceleration Requirement Relationships

Figure B.13 focuses on the Acceleration requirement, and relates it to other requirements and model elements. The “refine” relation, introduced in Figure B.12, shows how the Acceleration requirement is refined by a similarly named use case. The Power requirement is satisfied by the PowerSubsystem, and a Max Acceleration test case verifies the Acceleration requirement.

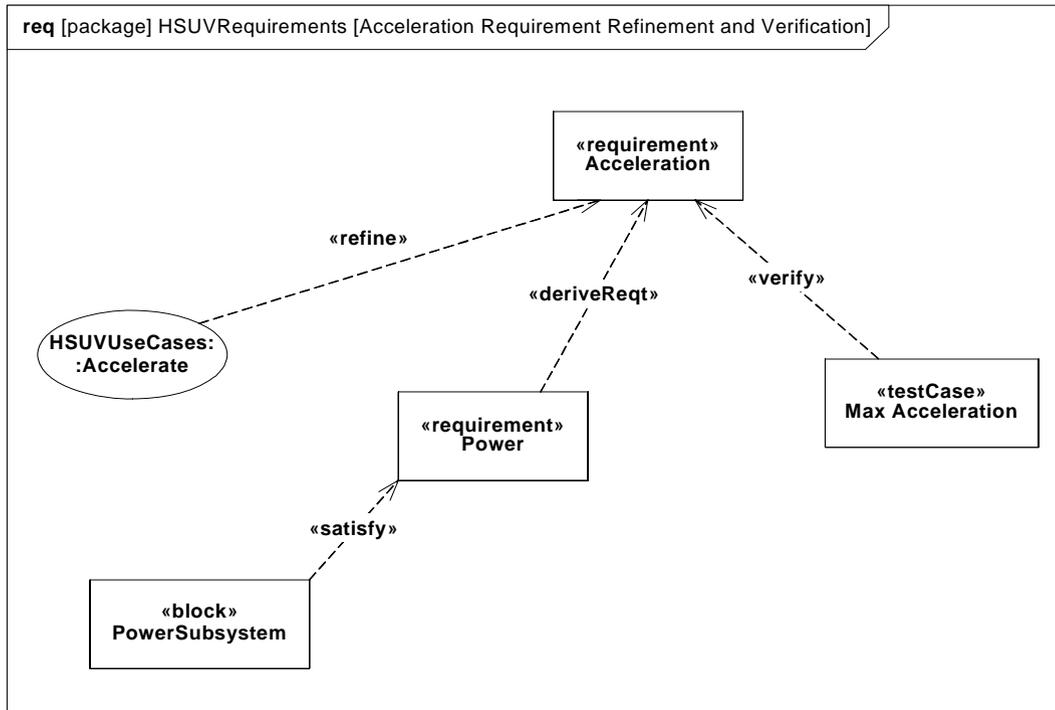


Figure B.13 - Acceleration Requirement Relationships (Requirements Diagram)

#### B.4.4.4 Table - Requirements Table

Figure B.14 contains two diagrams that show requirement containment (decomposition), and requirements derivation in tabular form. This is a more compact representation than the requirements diagrams shown previously.

table [requirement] Performance [Decomposition of Performance Requirement]		
id	name	text
2	Performance	The Hybrid SUV shall have the braking, acceleration, and off-road capability of a typical SUV, but have dramatically better fuel economy.
2.1	Braking	The Hybrid SUV shall have the braking capability of a typical SUV.
2.2	FuelEconomy	The Hybrid SUV shall have dramatically better fuel economy than a typical SUV.
2.3	OffRoadCapability	The Hybrid SUV shall have the off-road capability of a typical SUV.
2.4	Acceleration	The Hybrid SUV shall have the acceleration of a typical SUV.

table [requirement] Performance [Tree of Performance Requirements]							
id	name	relation	id	name	relation	id	name
2.1	Braking	deriveReq	d.1	RegenerativeBraking			
2.2	FuelEconomy	deriveReq	d.1	RegenerativeBraking			
2.2	FuelEconomy	deriveReq	d.2	Range			
4.2	FuelCapacity	deriveReq	d.2	Range			
2.3	OffRoadCapability	deriveReq	d.4	Power	deriveReq	d.2	PowerSourceManagement
2.4	Acceleration	deriveReq	d.4	Power	deriveReq	d.2	PowerSourceManagement
4.1	CargoCapacity	deriveReq	d.4	Power	deriveReq	d.2	PowerSourceManagement

Figure B.14 - Requirements Relationships Expressed in Tabular Format (Table)

## B.4.5 Breaking Down the Pieces (Block Definition Diagrams, Internal Block Diagrams)

### B.4.5.1 Block Definition Diagram - Automotive Domain

Figure B.15 provides definition for the concepts previously shown in the context diagram. Note that the interactions DriveBlackBox and StartVehicleBlackBox (described in Section B.4.3, “Elaborating Behavior (Sequence and State Machine Diagrams),” on page 166) are depicted as owned by the AutomotiveDomain block.

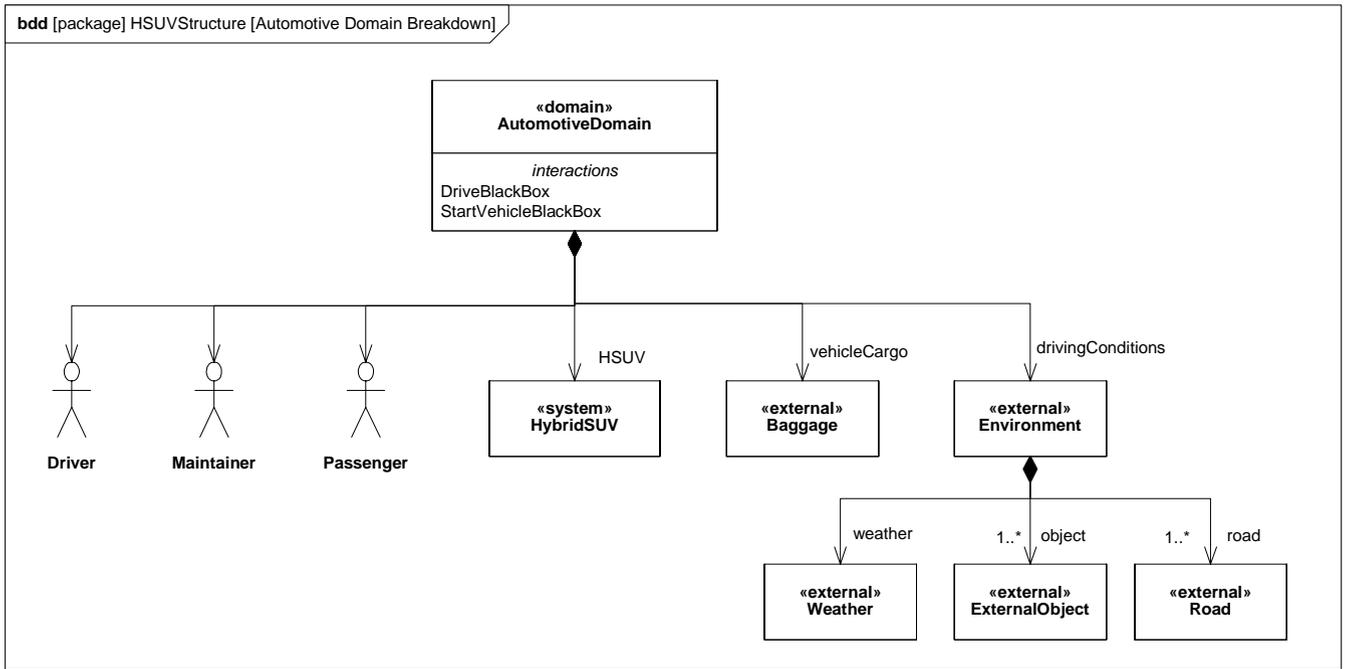


Figure B.15 - Defining the Automotive Domain (compare with Figure B.4 ) - (Block Definition Diagram)

### B.4.5.2 Block Definition Diagram - Hybrid SUV

Figure B.16 defines components of the HybridSUV block. Note that the BrakePedal and WheelHubAssembly are used by, but not contained in, the PowerSubsystem block.

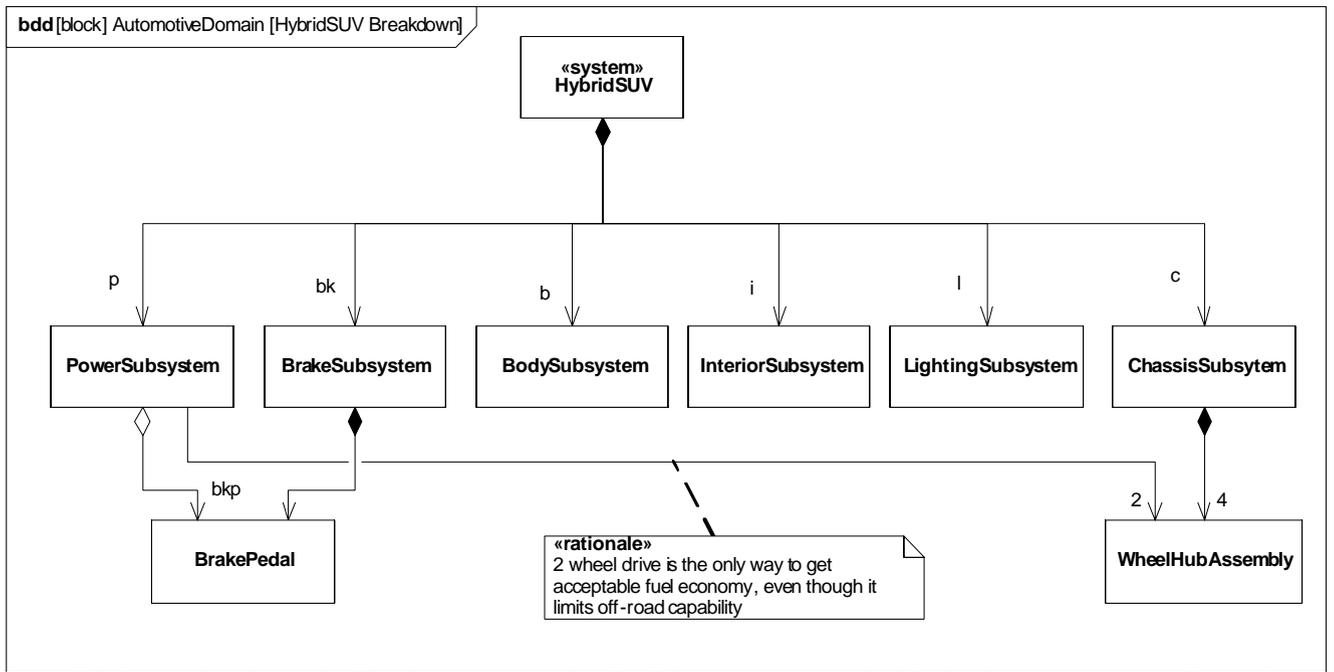


Figure B.16 - Defining Structure of the Hybrid SUV System (Block Definition Diagram)

### B.4.5.3 Internal Block Diagram - Hybrid SUV

Figure B.17 shows how the top level model elements in the above diagram are connected together in the HybridSUV block.

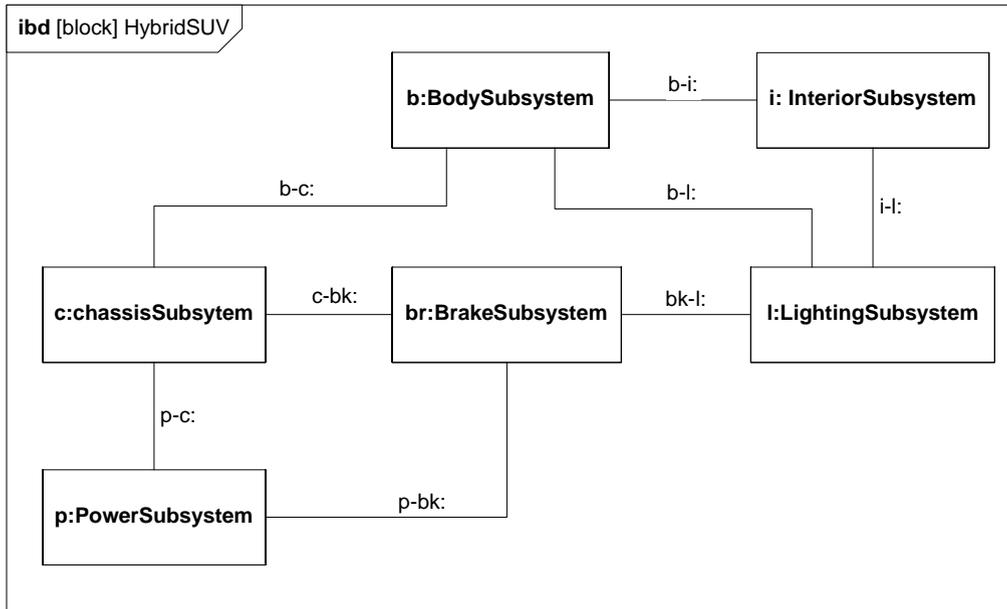


Figure B.17 - Internal Structure of Hybrid SUV (Internal Block Diagram)

### B.4.5.4 Block Definition Diagram - Power Subsystem

Figure B.18 defines the next level of decomposition, namely the components of the PowerSubsystem block. Note how the use of white diamond (shared aggregation) on FrontWheel, BrakePedal, and others denotes the same “use-not-composition” kind of relationship previously shown in Figure B.16.

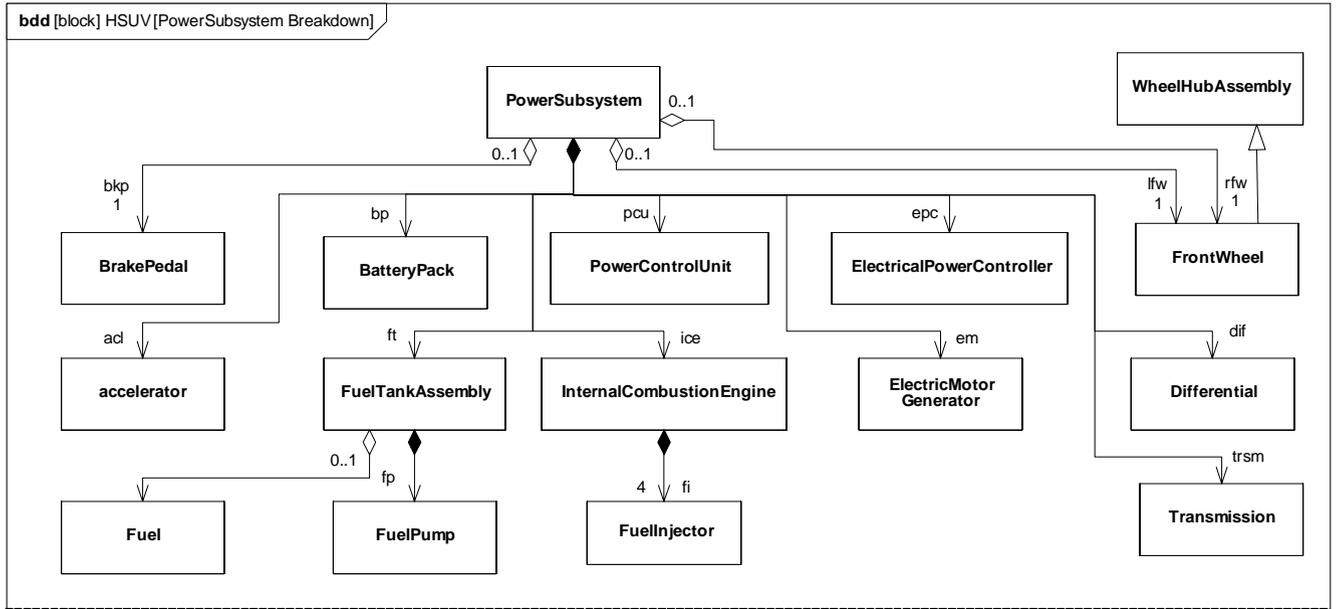


Figure B.18 - Defining Structure of Power Subsystem (Block Definition Diagram)

### B.4.5.5 Internal Block Diagram for the “Power Subsystem”

Figure B.19 shows how the parts of the PowerSubsystem block, as defined in the diagram above, are used. It shows «connectors» between parts, «clientServerPorts», «flowPorts», «atomicFlowPorts», and «itemFlows». The dashed borders on FrontWheel and BrakePedal denote the “use-not-composition” relationship depicted elsewhere in Figure B.16 and Figure B.18. The dashed borders on Fuel denote a store, which keeps track of the amount and mass of fuel in the FuelTankAssy. This is also depicted in Figure B.18.

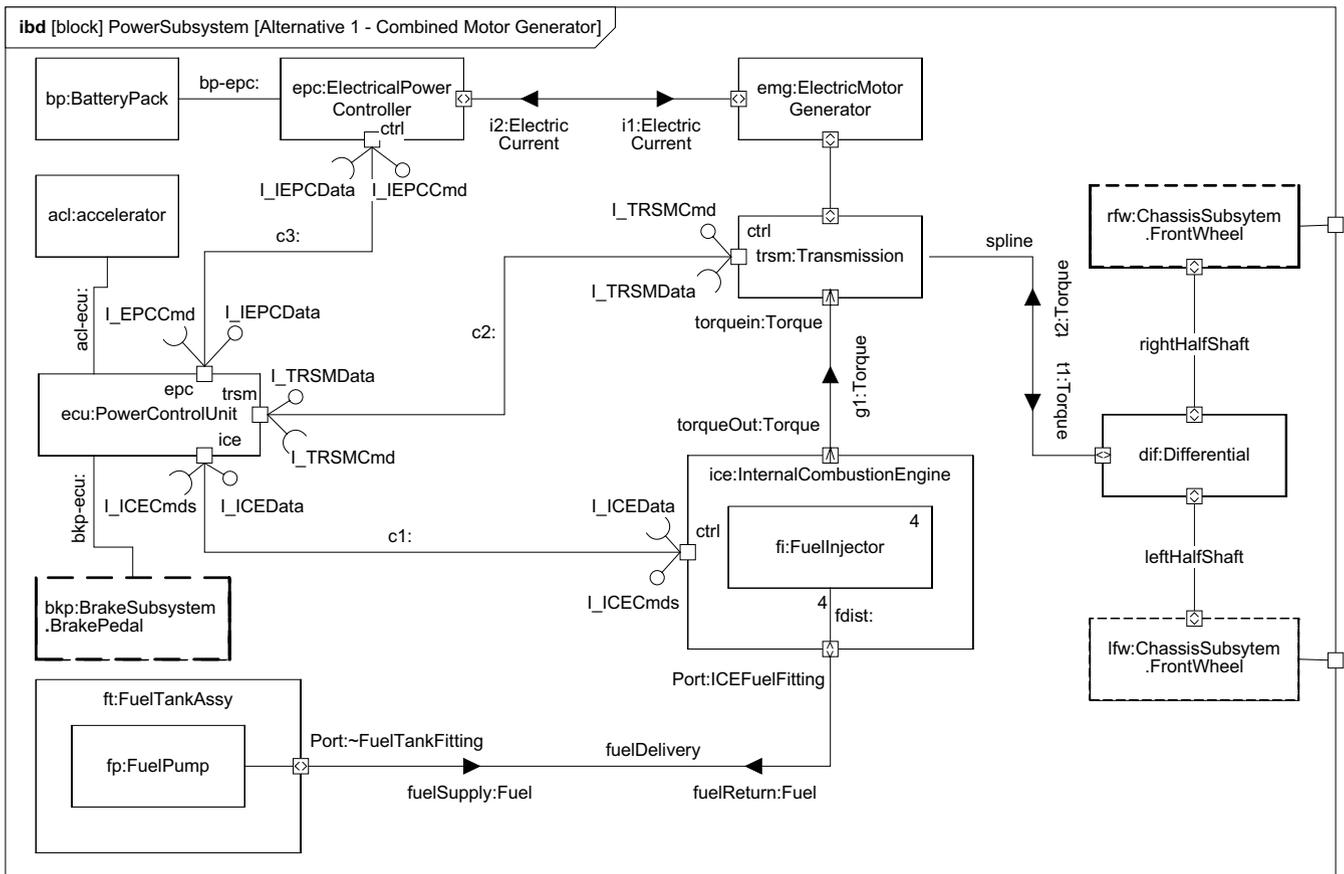


Figure B.19 - Internal Structure of the Power Subsystem (Internal Block Diagram)

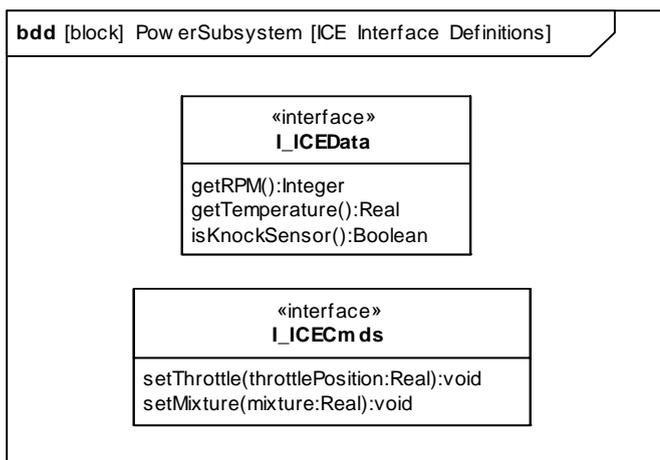


Figure B.20 - Interfaces Typing StandardPorts Internal to the Power Subsystem (Block Definition Diagram)

Figure B.20 provides definition of the interfaces applied to Standard Ports associated with connector c1 in Figure B.19.

## B.4.6 Defining Ports and Flows

### B.4.6.1 Block Definition Diagram - ICE Interface

For purposes of example, the StandardPorts and related point-to-point connectors in Figure B.19 are being refined into a common bus architecture. For this example, FlowPorts have been used to model the bus architecture. Figure B.21 is an incomplete first step in the refinement of this bus architecture, as it begins to identify the flow specification for the InternalCombustionEngine, the Transmission, and the ElectricalPowerController..

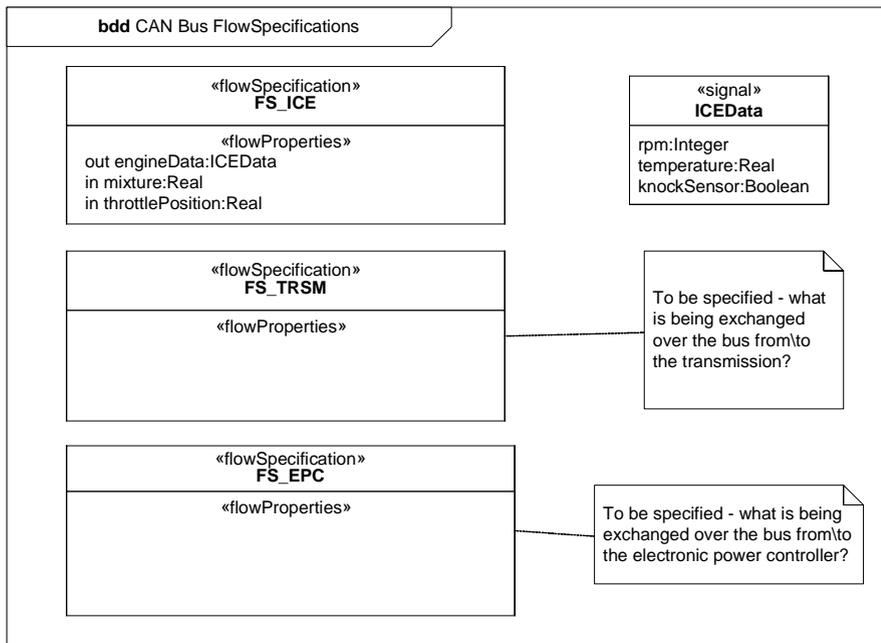


Figure B.21 - Initially Defining Flow Specifications for the CAN Bus (Block Definition Diagram)

### B.4.6.2 Internal Block Diagram - CANbus

Figure B.22 continues the refinement of this Controller Area Network (CAN) bus architecture using FlowPorts. The explicit structural allocation between the original connectors of Figure B.19 and this new bus architecture is shown in Figure B.36.

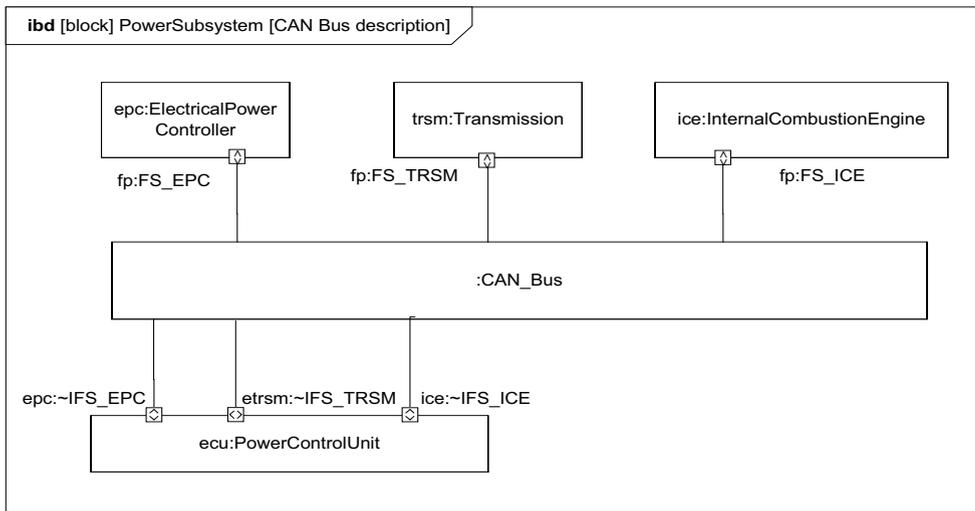


Figure B.22 - Consolidating Interfaces into the CAN Bus. (Internal Block Diagram)

### B.4.6.3 Block Definition Diagram - Fuel Flow Properties

The FlowPorts on the FuelTankAssembly and InternalCombustionEngine (as shown in Figure B.19) are defined in Figure B.23.

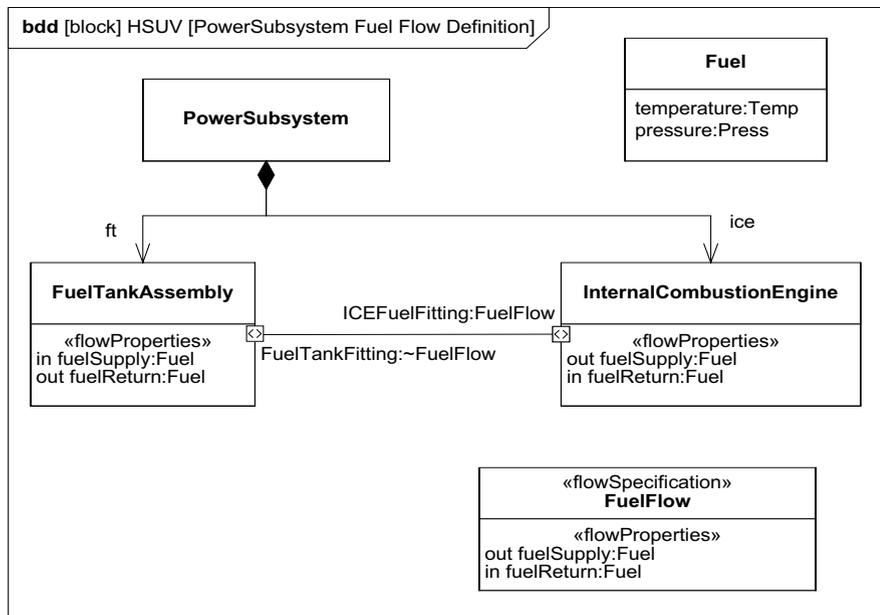


Figure B.23 - Elaborating Definition of Fuel Flow. (Block Definition Diagram)

#### B.4.6.4 Parametric Diagram - Fuel Flow

Figure B.24 is a parametric diagram showing how fuel flowrate is related to FuelDemand and FuelPressure value properties.

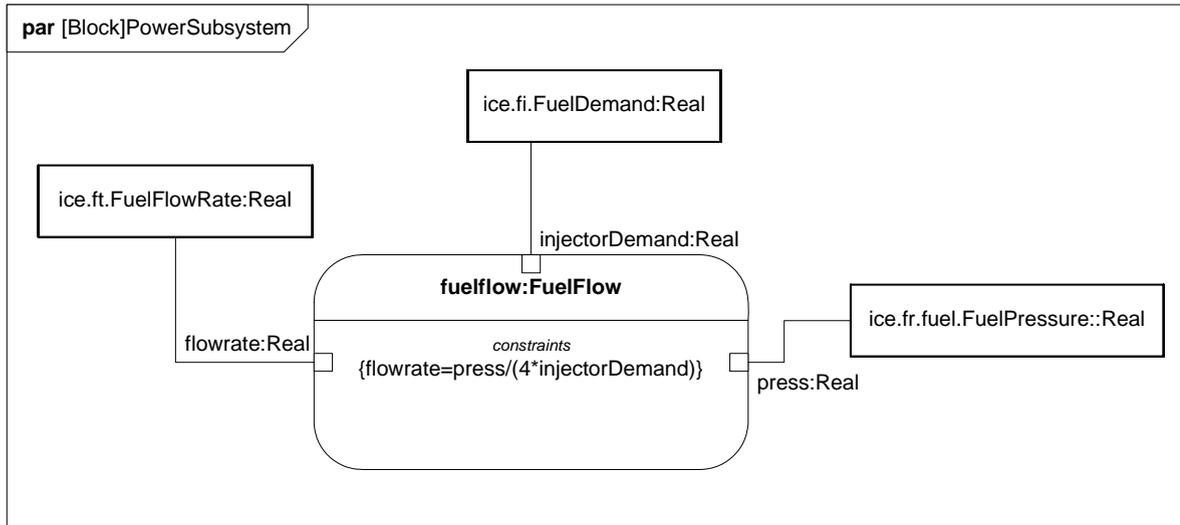


Figure B.24 - Defining Fuel Flow Constraints (Parametric Diagram)

#### B.4.6.5 Internal Block Diagram - Fuel Distribution

Figure B.25 shows how the connectors fuelDelivery and fdist on Figure B.19 have been expanded to include design detail. The fuelDelivery connector is actually two connectors, one carrying fuelSupply and the other carrying fuelReturn. The fdist connector inside the InternalCombustionEngine block has been expanded into the fuel regulator and fuel rail parts. These more detailed design elements are related to the original connectors using the allocation relationship. The Fuel store represents a quantity of fuel in the FuelTankAssy, which is drawn by the FuelPump for use in the engine, and is refreshed, to some degree, by fuel returning to the FuelTankAssy via the FuelReturnLine.

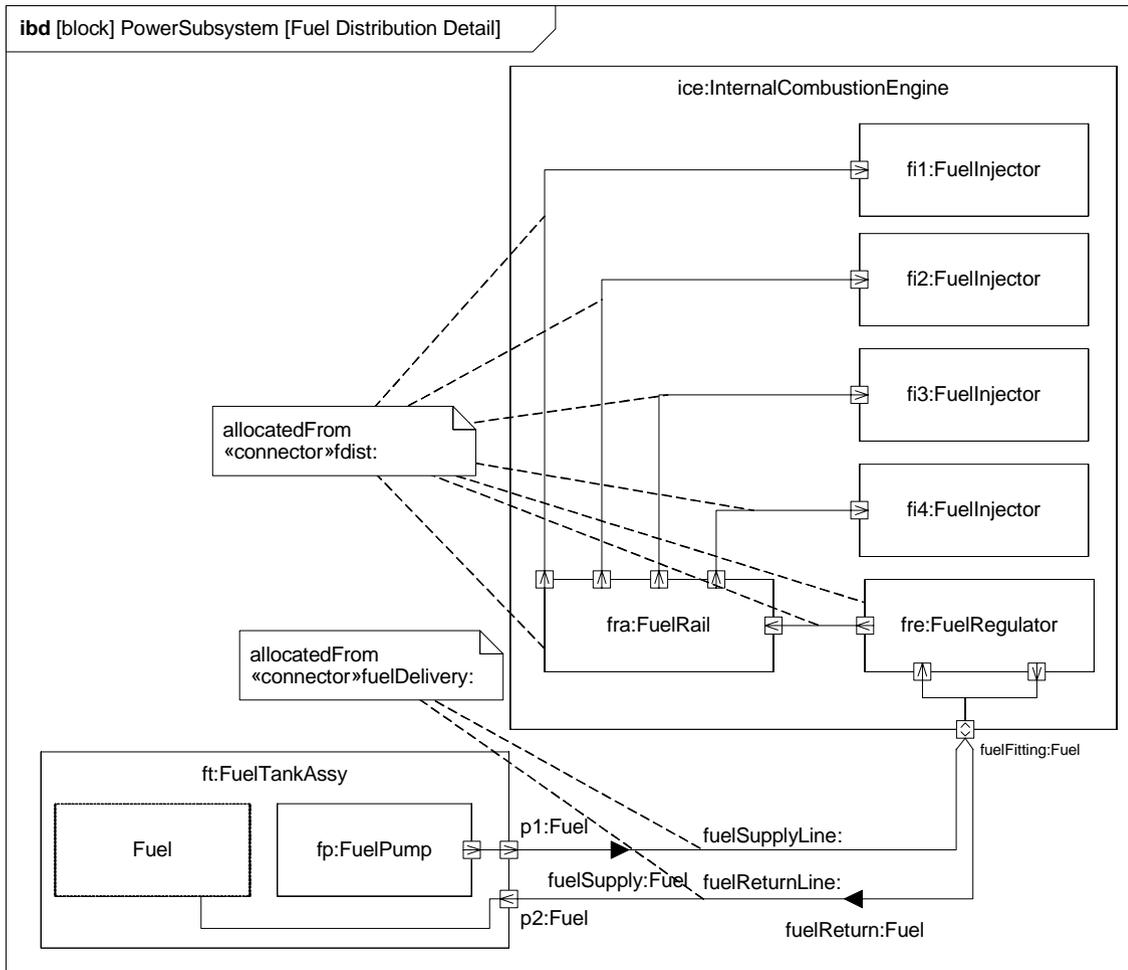


Figure B.25 - Detailed Internal Structure of Fuel Delivery Subsystem (Internal Block Diagram)

## B.4.7 Analyze Performance (Constraint Diagrams, Timing Diagrams, Views)

### B.4.7.1 Block Definition Diagram - Analysis Context

Figure B.26 defines the various model elements that will be used to conduct analysis in this example. It depicts each of the constraint blocks/equations that will be used for the analysis, and key relationships between them.

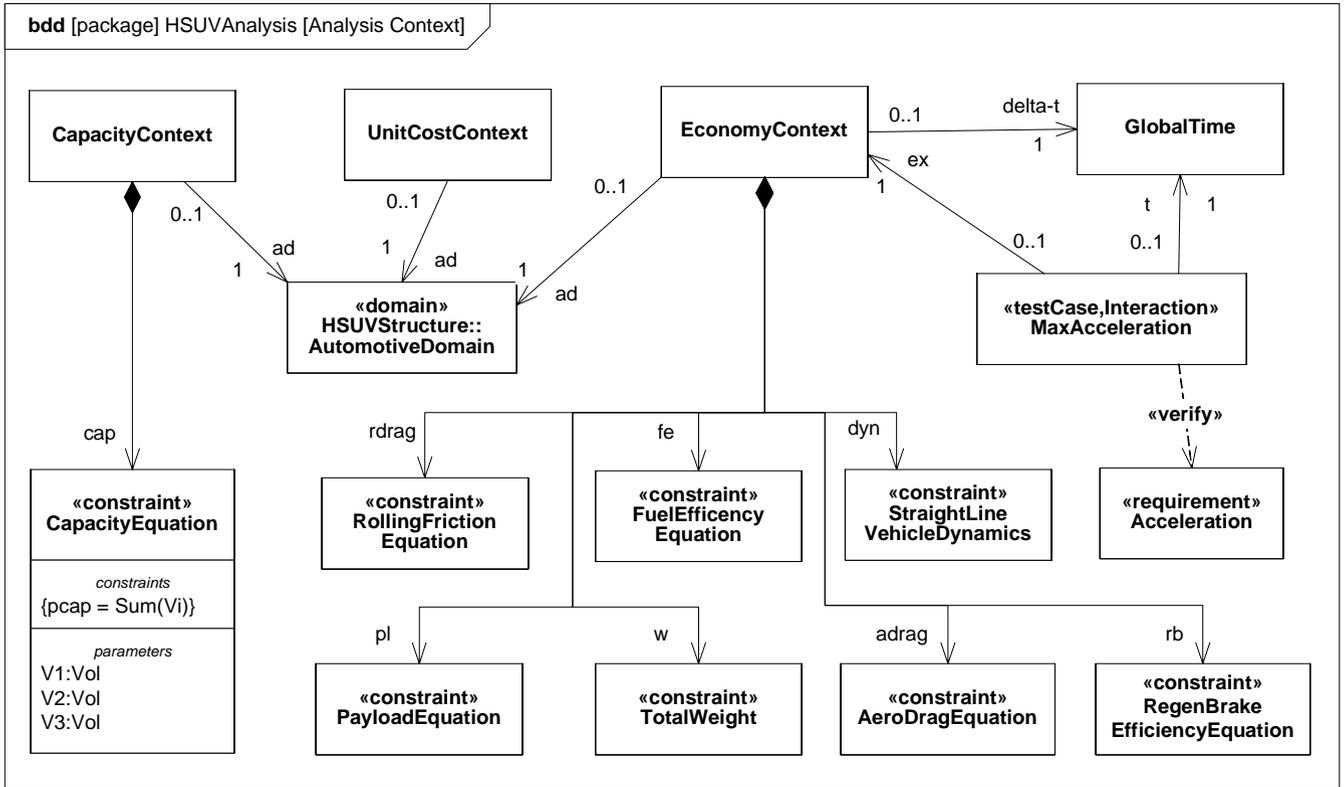


Figure B.26 - Defining Analyses for Hybrid SUV Engineering Development (Block Definition Diagram)

### B.4.7.2 Package Diagram - Performance View Definition

Figure B.27 shows the user-defined Performance Viewpoint, and the elements that populate the HSUV specific PerformanceView. The PerformanceView itself may contain of a number of diagrams depicting the elements it contains.

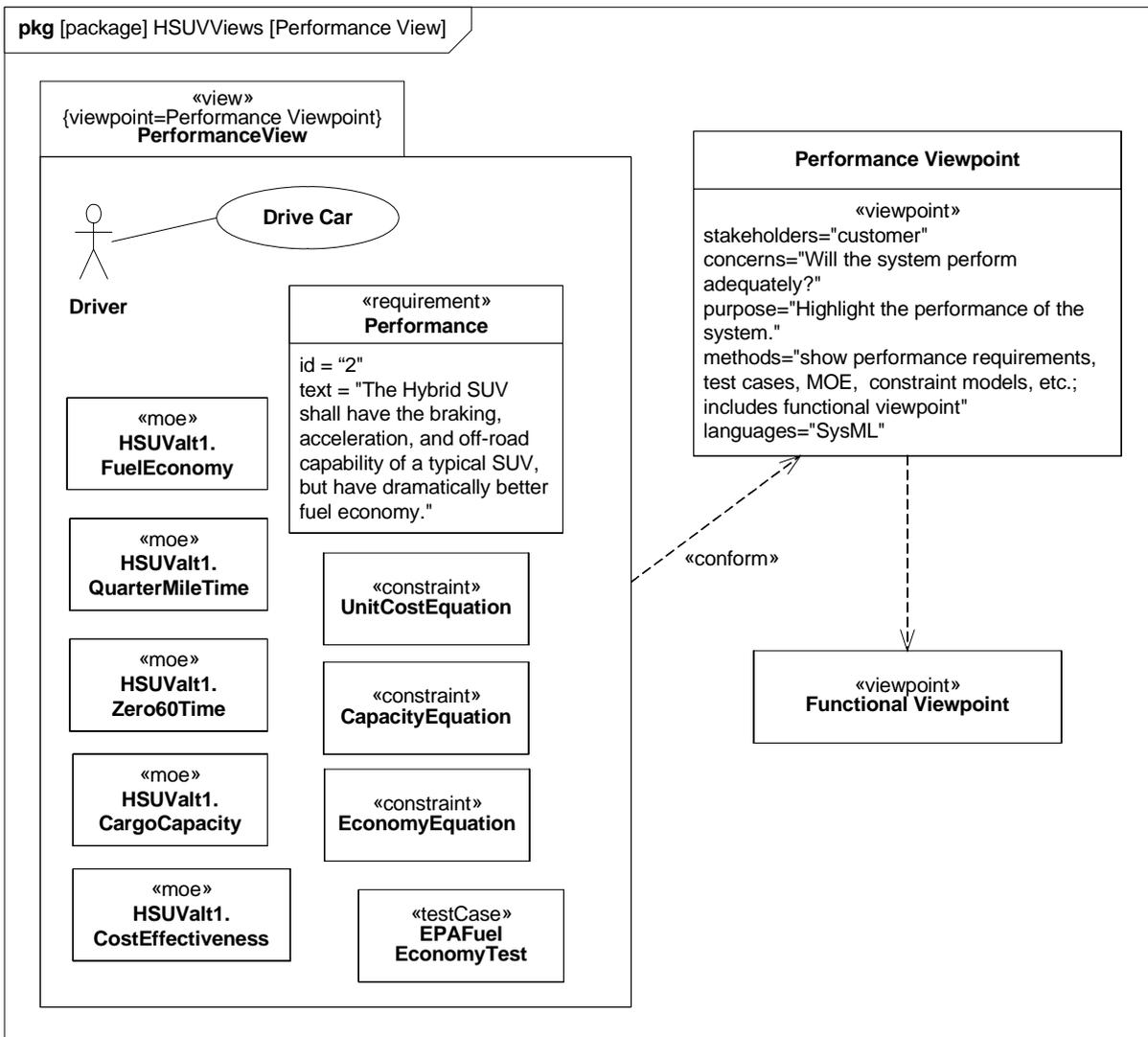


Figure B.27 - Establishing a Performance View of the User Model (Package Diagram)

### B.4.7.3 Parametric Diagram - Measures of Effectiveness

Measure of Effectiveness is a user defined stereotype. Figure B.28 shows how the overall cost effectiveness of the HSUV will be evaluated. It shows the particular measures of effectiveness for one particular alternative for the HSUV design, and can be reused to evaluate other alternatives.

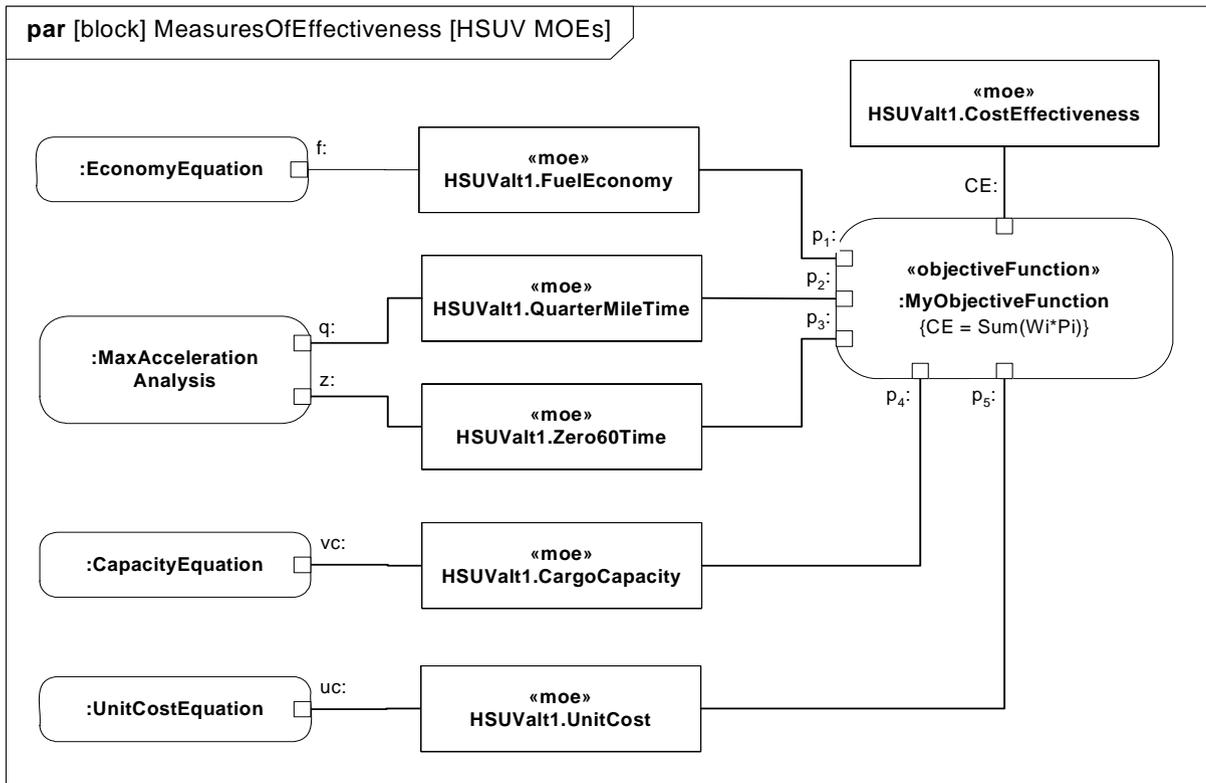
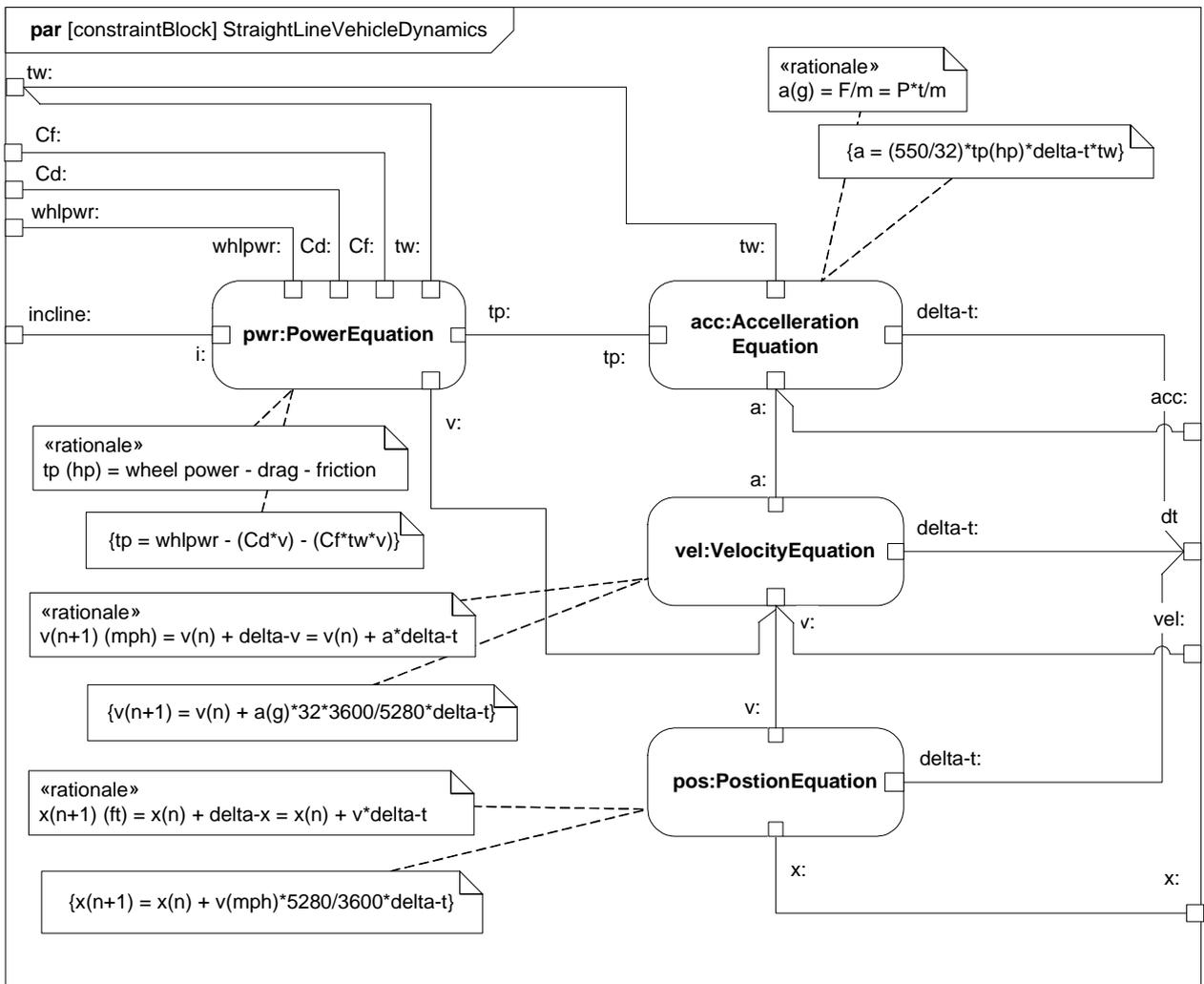


Figure B.28 - Defining Measures of Effectiveness and Key Relationships (Parametric Diagram)

#### B.4.7.4 Parametric Diagram - Economy

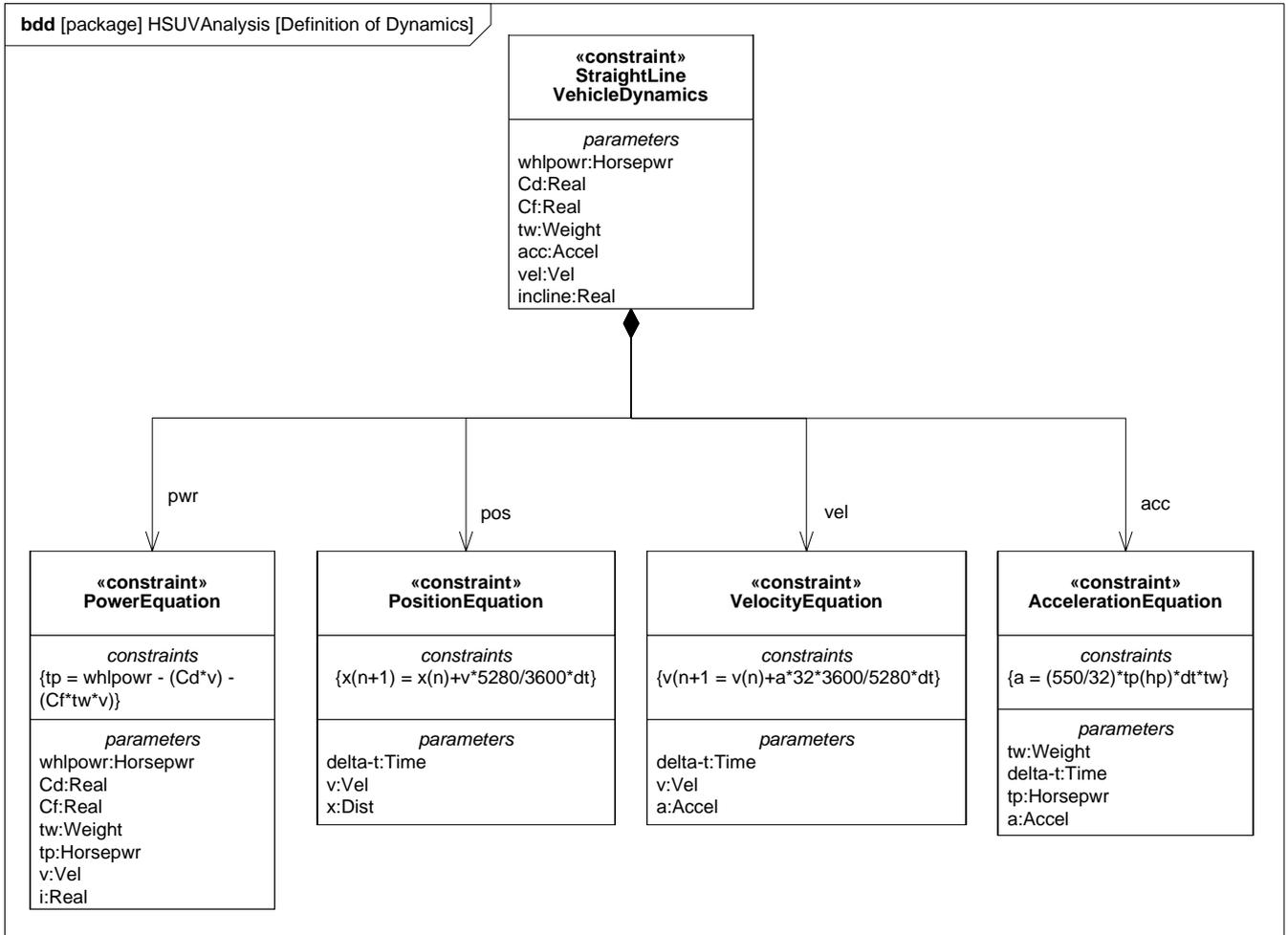
Since overall fuel economy is a key requirement on the HSUV design, this example applies significant detail in assessing it. Figure B.29 shows the constraint blocks and properties necessary to evaluate fuel economy.





**Figure B.30 - Straight Line Vehicle Dynamics Mathematical Model (Parametric Diagram)**

The constraints and parameters in Figure B.30 are detailed in Figure B.31 in Block Definition Diagram format.



**Figure B.31 - Defining Straight-Line Vehicle Dynamics Mathematical Constraints (Block Definition Diagram)**

Note the use of valueTypes originally defined in Figure B.2.

#### B.4.7.6 (Non-Normative) Timing Diagram - 100hp Acceleration

Timing diagrams, while included in UML 2, are not directly supported by SysML. For illustration purposes, however, the interaction shown in Figure B.32 was generated based on the constraints and parameters of the StraightLineVehicleDynamics constraintBlock, as described in the Figure B.30. It assumes a constant 100hp at the drive wheels, 4000lb gross vehicle weight, and constant values for Cd and Cf.

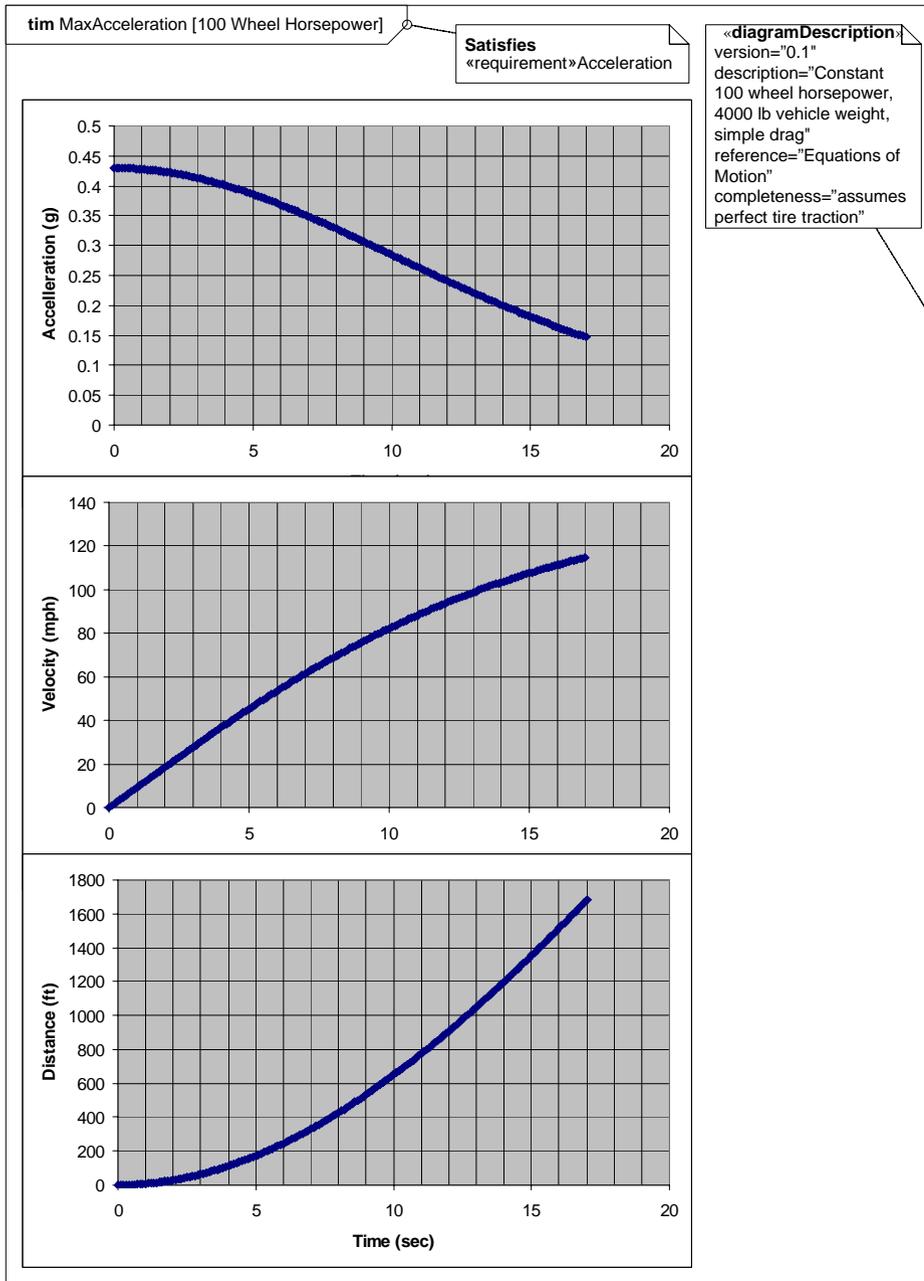


Figure B.32 - Results of Maximum Acceleration Analysis (Timing Diagram)

## B.4.8 Defining, Decomposing, and Allocating Activities

### B.4.8.1 Activity Diagram - Acceleration (top level)

Figure B.33 shows the top level behavior of an activity representing acceleration of the HSUV. It is the intent of the systems engineer in this example to allocate this behavior to parts of the PowerSubsystem. It is quickly found, however, that the behavior as depicted cannot be allocated, and must be further decomposed.

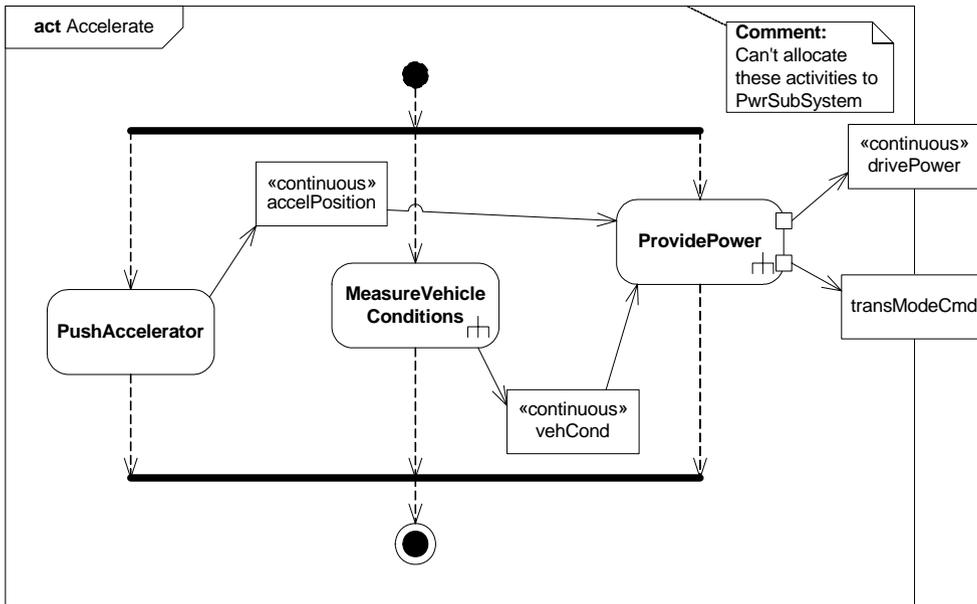


Figure B.33 - Behavior Model for "Accelerate" Function (Activity Diagram)

### B.4.8.2 Block Definition Diagram - Acceleration

Figure B.34 defines a decomposition of the activities and objectFlows from the activity diagram in Figure B.33.

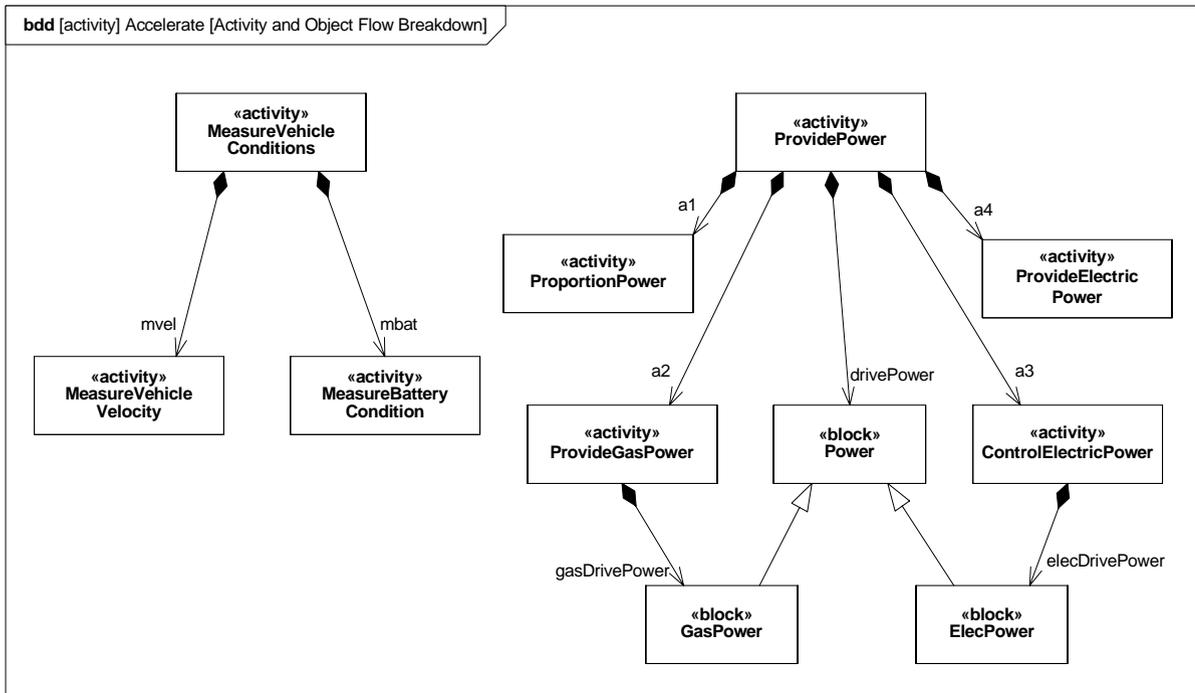
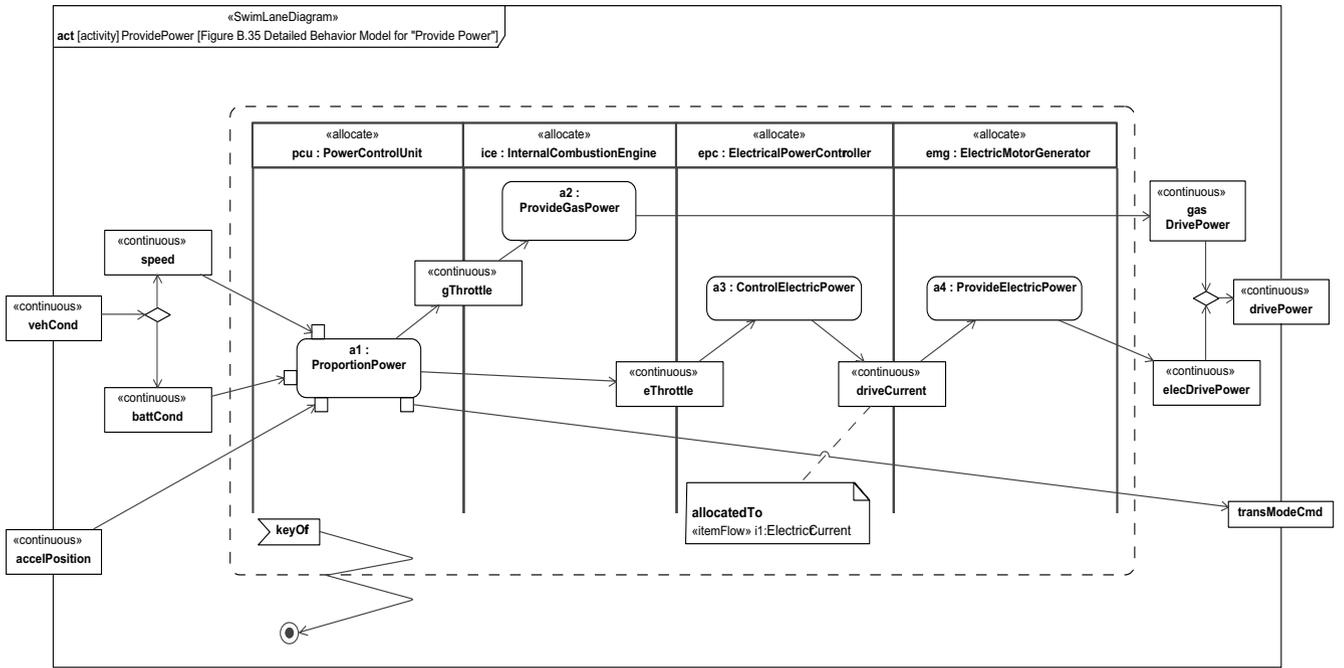


Figure B.34 - Decomposition of “Accelerate” Function (Block Definition diagram)

### B.4.8.3 Activity Diagram (EFFBD) - Acceleration (detail)

Figure B.35 shows the ProvidePower activity, which includes Actions invoking the decomposed Activities and ObjectNodes from Figure B.34. It also uses AllocateActivityPartitions and an allocation callout to explicitly allocate activities and an object flow to parts in the PowerSubsystem block.

Note that the incoming and outgoing object flows for the ProvidePower activity have been decomposed. This was done to distinguish the flow of electrically generated mechanical power and gas generated mechanical power, and to provide further insight into the specific vehicle conditions being monitored.



**Figure B.35 - Detailed Behavior Model for "Provide Power" (Activity Diagram)**  
 Note hierarchical consistency with Figure B.33.

### B.4.8.4 Internal Block Diagram - Power Subsystem Behavioral and Flow Allocation

Figure B.36 depicts a subset of the PowerSubsystem, specifically showing the allocation relationships generated in Figure B.35.

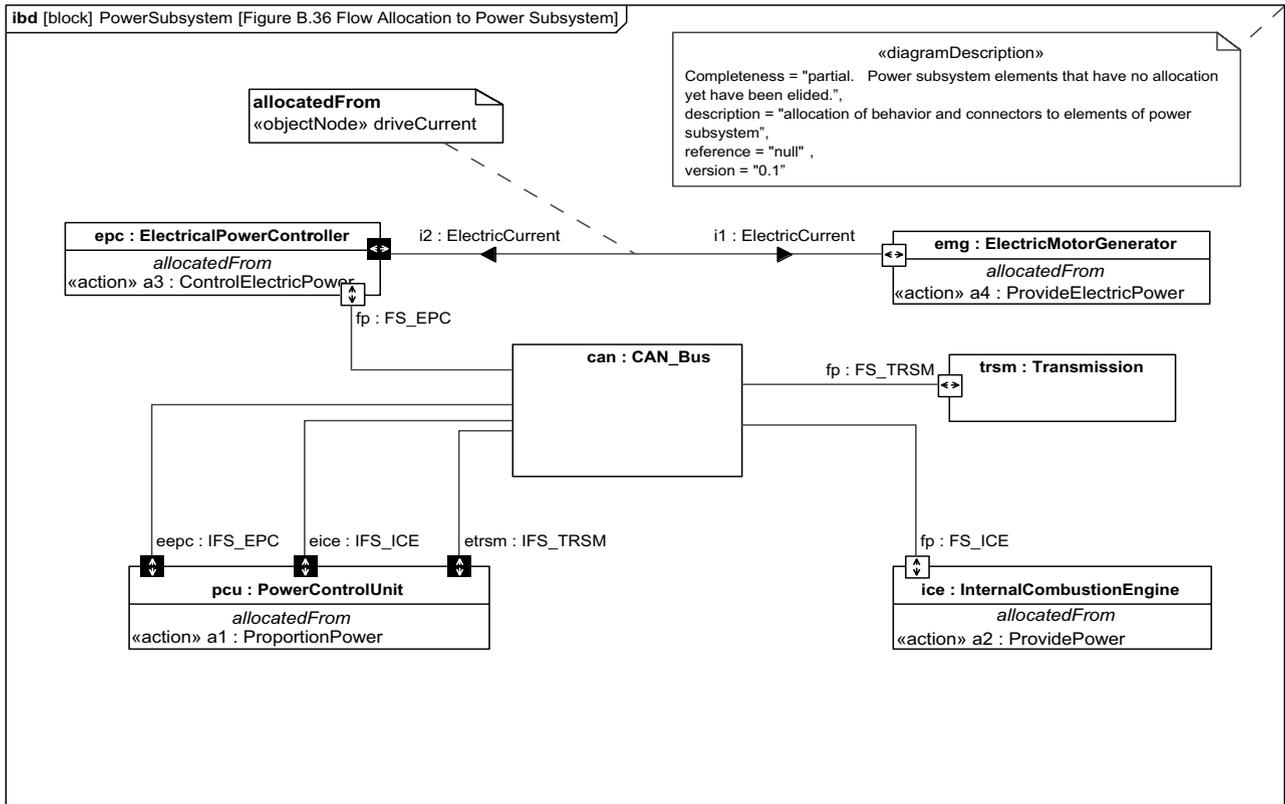


Figure B.36 - Flow Allocation to Power Subsystem (Internal Block Diagram)

### B.4.8.5 Table - Acceleration Allocation

Figure B.37 shows the same allocation relationships shown in Figure B.36, but in a more compact tabular representation.

Type	Name	End	Relation	End	Type	Name
action	a1 : ProportionPower	from	allocate	to	part	ecu : PowerControlUnit
action	a2 : ProvideGasPower	from	allocate	to	part	ice : InternalCombustionEngine
action	a3 : ControlElectricPower	from	allocate	to	part	epc : ElectricPowerController
action	a4 : ProvideElectricPower	from	allocate	to	part	emg : ElectricMotorGenerator
objectFlow	o6	from	allocate	to	connector	epc-emg.1

Figure B.37 - Tabular Representation of Allocation from "Accelerate" Behavior Model to Power Subsystem (Table)

### B.4.8.6 Internal Block Diagram: Property Specific Values - EPA Fuel Economy Test

Figure B.38 shows a particular Hybrid SUV (VIN number) satisfying the EPA fuel economy test. Serial numbers of specific relevant parts are indicated.

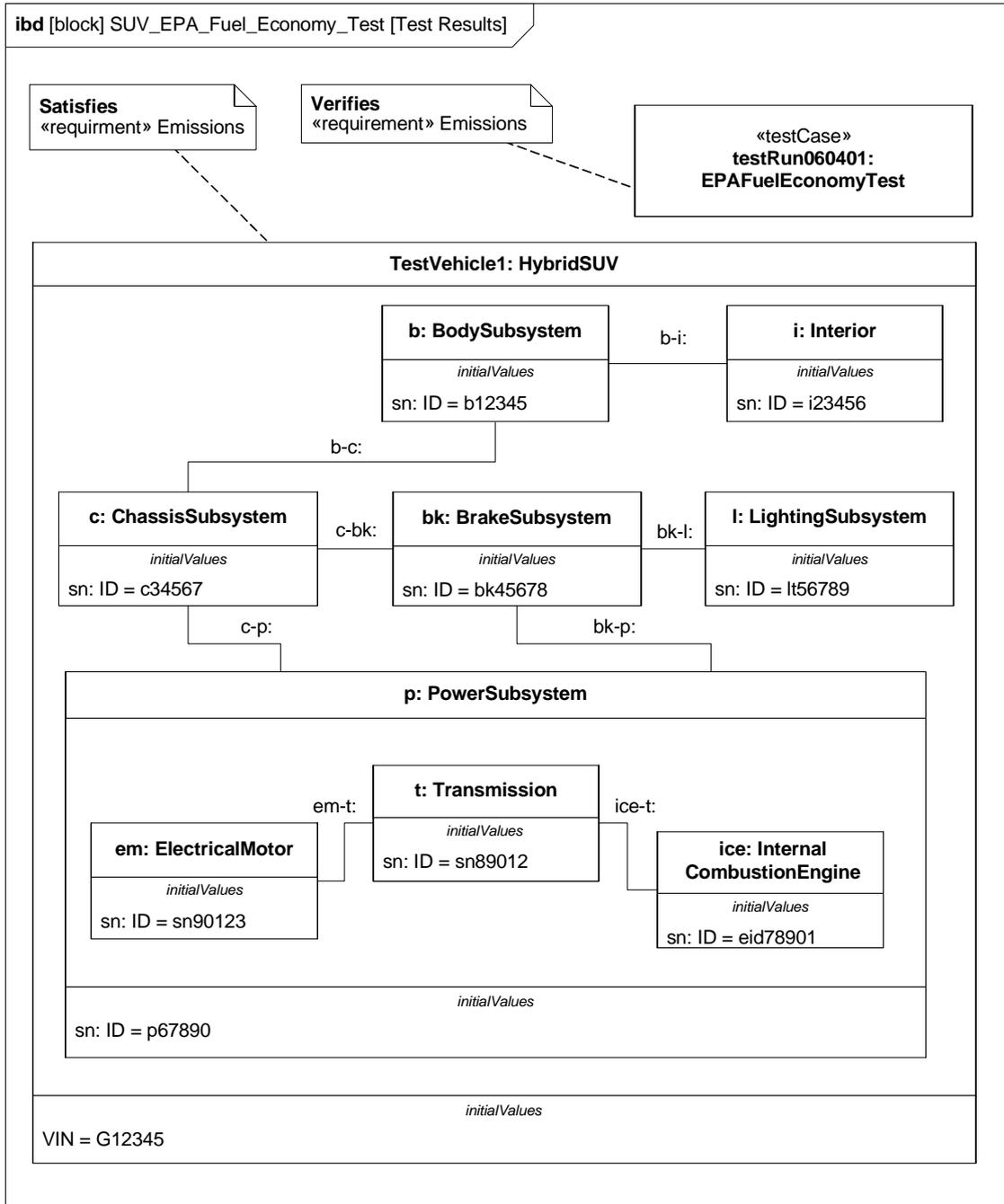


Figure B.38 - Special Case of Internal Block Diagram Showing Reference to Specific Properties (serial numbers)

